

Université Aboubekr Belkaïd - Tlemcen
Laboratoire d'Automatique - Équipe de Productique

Mémoire de Magister "API" (Automatique Productique Informatique)
Option "Productique"

Présenté par

BESSENOUCI Hakim Nadhir

**Pilotage d'un AS/RS à convoyeur gravitationnel
par Métaheuristique**

Soutenue le ??????????
Devant la Commission d'Examen

JURY

M. Sari Zaki	Professeur d'Université, FT-UABT	Encadreur
Melle. Latéfa GHOMRI	Maître Assistante, FT - UABT	Co-encadreur
M. Melani Sidi Mohamed	Maître de conférence(B), FT-UABT	Examineur
M. Cheikh Abdelmadjid	Professeur d'Université, FT-UABT	Président
M.	xxxxxxxxxx	examineur

Remerciements

Qu'il nous soit permis, en cette heureuse occasion, d'exprimer notre reconnaissance et notre profonde gratitude à tous ceux et celles qui nous ont faits, qui nous ont formés et qui ont suivi patiemment nos pas jusqu'à la réalisation de cette étape, concrétisée par ce modeste travail. A ceux-là, nous adresserons nos remerciements les plus chaleureux pour les efforts qu'ils n'ont cessé de nous prodiguer tout au long de ce périple sinueux et bien difficile.

Il s'agit bien sûr de nos parents qui nous ont vu naître et grandir et qui, bien souvent, se sont privés pour satisfaire nos exigences et notre désir d'apprendre.

Il s'agit également de nos enseignants qui nous ont inculqué les bases de la science et les principes de la vie.

Il s'agit, surtout, de nos encadreurs Mr Zaki SARI et Melle. Latéfa GHOMRI qui, malgré leurs lourdes et bien nombreuses charges, n'ont pas hésité à puiser de leur précieux temps pour nous orienter dans notre travail et nous entourer de leurs précieux enseignements.

Il s'agit, encore, de tous ceux qui nous ont ouvert les portes de leurs institutions pour nous permettre l'accès à la connaissance quelle qu'elle soit et où qu'elle soit.

Nous ne saurions oublier, enfin, de remercier respectueusement les membres de notre jury d'examen qui ont accepté d'examiner cet humble travail que nous espérons à la hauteur de nos ambitions bien modestes.

Il s'agit de MM :

M. Cheikh Abdelmadjid

M. Abderrahim Amine

M. Melani Sidi Mohamed

A tous ceux-là, nous demeurerons biens obligés.

Table des matières

Table des figures	1
Liste des tableaux	3
Introduction générale	4
I Introduction aux Systèmes Automatisés de Stockage/Déstockage.	8
1 Introduction	8
2 Système automatisé de stockage/déstockage AS/RS	9
2.1 Définition	9
2.2 Fonction	9
2.3 Composition	9
2.4 Terminologie	9
3 Types d'AS/RS	10
3.1 Les AS/RS à charge unitaire	10
3.2 Les AS/RS multi allées	11
3.3 Les AS/RS à racks glissants	12
3.4 Les AS/RS à mini charge	12
3.5 Les AS/RS à carrousel	12
3.6 Les AS/RS à personne embarquée	14
3.7 Les AS/RS à personne embarquée	14
3.8 Les AS/RS à convoyeur gravitationnel	15
4 Conception d'un AS/RS	16
5 Méthodologies de stockage	16
5.1 Stockage dédié	17
5.2 Stockage aléatoire ouvert	17
5.3 Stockage aléatoire par classe	17
6 Disposition de l'aire de stockage	18
6.1 Le Bâtiment	18

6.2	Les Allées	18
6.3	Les étagères	18
7	L'évaluation des performances des AS/RS dans la littérature	19
8	Conclusion	20
II Systèmes Automatisés de Stockage/Déstockage à Convoyeur Gravitationnel		21
1	Introduction	21
2	Système AS/RS à Convoyeur Gravitationnel	21
2.1	Composantes	22
2.2	Fonctionnement du système AS/R à Convoyeur Gravitationnel	23
2.3	Modélisation des temps de transport d'un AS/RS à Convoyeur Gravitationnel	24
3	Heuristique pour le stockage/déstockage dans un AS/RS à convoyeur gravitationnel	28
3.1	Procédure de stockage	29
3.2	Procédure de déstockage	32
4	Conclusion	33
III Les Méthaheuristiques		34
1	Introduction	34
2	Méthaheuristiques	34
2.1	Définition	34
2.2	Formulation du problème	35
2.3	propriétés fondamentales des méthaheuristiques	35
2.4	classifications des méthaheuristiques [Her05]	36
3	Méthode du recuit simulé (simulated annealing)[JD03]	37
3.1	Analogie avec la thermodynamique	37
3.2	Algorithme du recuit simulé	38
3.3	Réglage des paramètres de la méthode du Recuit Simulé	39
4	La Méthode de recherche avec tabous (Tabu Search)[JD03]	40
4.1	Principe de la méthode Tabou	41
4.2	Algorithme de la recherche Tabou	41
4.3	Améliorations de la méthode de la recherche tabou	41
5	Conclusion	43
IV Simulateur d'AS/RS à convoyeur gravitationnel		44
1	Introduction	44
2	Introduction à la simulation	44
2.1	Définition	44
2.2	Méthodologie d'une simulation	44
2.3	Avantages et inconvénients de la simulation	45
2.4	Simulation de systèmes à événements discrets	46
2.5	Logiciels de simulation	46
3	Modélisation du Système AS/RS à convoyeur gravitationnel	47

3.1	description des principales classes	50
4	Implémentation	51
4.1	JAVA	51
4.2	Environnement de développement ECLIPSE	52
5	Validation du simulateur	52
6	Conclusion	54
V	Adaptation du recuit simulé et la recherche tabou pour l'AS/RS à convoyeur gravitationnel	55
1	Introduction	55
2	Les algorithmes des métaheuristiques	55
3	Etude de sensibilité	56
4	Analyse et comparaison des résultats de simulation	59
	Conclusion générale et perspectives	67
	Publication Internationale	69
	Références bibliographiques	85

Table des figures

I.1	AS/RS à charge unitaire.	11
I.2	AS/RS multi allées.	11
I.3	AS/RS à racks glissants.	12
I.4	AS/RS à mini charge	13
I.5	AS/RS à carrousel horizontal	13
I.6	AS/RS à carrousel vertical	14
I.7	AS/RS à personne embarquée	15
I.8	AS/RS à étagères profondes	15
I.9	AS/RS à convoyeur gravitationnel	16
II.1	Configuration AS/RS à Convoyeur Gravitationnel	22
III.1	Fonction objectif d'un problème d'optimisation difficile en fonction de la configuration [JD03]	38
III.2	comparaison des technique du recuit et de la trempe[JD03]	38
III.3	Organigramme de l'algorithme du recuit simulé	40
III.4	Organigramme de la Recherche avec tabou	42
IV.1	Méthodologie d'une simulation	45
IV.2	Diagramme des classes	49
IV.3	Temps moyen de déstockage d'un AS/RS à convoyeur gravitationnel en fonction du taux de charge	53
V.1	temps de destockage des différentes méthodes par rapport au nombre de type produit pour un stockage aléatoire et un lot de taille 2	60
V.2	temps de destockage des différentes méthodes par rapport au nombre de type produit pour un stockage avec heuristique et un lot de taille 2	62
V.3	temps de destockage des différentes méthodes par rapport au nombre de type produit pour un stockage aléatoire et un lot de taille 5	62

V.4	temps de destockage des différentes méthodes par rapport au nombre de type produit pour un stockage avec heuristique et un lot de taille 5	63
V.5	temps de destockage des différentes méthodes par rapport au nombre de type produit pour un stockage aléatoire et un lot de taille 200	64
V.6	temps de destockage des différentes méthodes par rapport au nombre de type produit pour un stockage avec heuristique et un lot de taille 200	65

Liste des tableaux

IV.1	Temps moyen de déstockage de différent configuration d'AS/RS à convoyeur gravitationnel	53
IV.2	Temps moyen de déstockage d'un AS/RS à convoyeur gravitationnel	53
V.1	résultat de l'étude de sensibilité du recuit simulé	57
V.2	résultat de l'étude de sensibilité de la recherche tabou	58
V.3	temps de destockage des différentes méthodes par rapport au nombre de type produit pour un stockage aléatoire et un lot de taille 2	59
V.4	temps de destockage des différentes méthodes par rapport au nombre de type produit pour un stockage avec heuristique et un lot de taille 2	61
V.5	temps de destockage des différentes méthodes par rapport au nombre de type produit pour un stockage aléatoire et un lot de taille 5	61
V.6	temps de destockage des différentes méthodes par rapport au nombre de type produit pour un stockage avec heuristique et un lot de taille 5	61
V.7	temps de destockage des différentes méthodes par rapport au nombre de type produit pour un stockage aléatoire et un lot de taille 200	63
V.8	temps de destockage des différentes méthodes par rapport au nombre de type produit pour un stockage avec heuristique et un lot de taille 200	64

Introduction générale

La globalisation des marchés ainsi que la compétition portée au niveau mondial ont imposé aux entreprises de miser de plus en plus sur la souplesse de leur fonction de production pour se démarquer de la concurrence.

Les pressions toujours grandissantes de réduction des délais de livraison et de la taille des lots de livraison obligent ces dernières à opter pour de nouveaux systèmes de production, tels que les systèmes flexibles de production (Flexible Manufacturing Systems, FMS). Ces systèmes flexibles de production permettent d'améliorer la productivité et d'accroître la rentabilité dans des conditions de production plus diversifiées en grandes et petites séries ; leurs structures leur permettent de produire un très large éventail de produits et donc d'être moins sujets à un crash économique.

Les systèmes flexibles de production (Flexible Manufacturing Systems, FMS) peuvent changer rapidement de produits ou de séquences de production, sans perdre leurs productivités, à condition qu'il soit possible d'obtenir le bon produit, palette, support, ou outil, à la bonne place et au bon moment. Ils se composent essentiellement : des stations de travail (par exemple, machines-outils, stations d'assemblage, robots, etc.), d'équipements de manutention et de transport (par exemple, robots, convoyeurs, véhicules autoguidés ...etc.), d'un système automatisé de stockage/déstockage, d'un système de communication et d'un système de gestion par ordinateur.

La flexibilité d'un système de production se caractérise par sa capacité d'adaptation à la production de nouveaux produits pour lesquels le système n'a pas été conçu. Cela suppose une adaptation totale du système de production au produit courant (de la distribution des flux discrets de composants aux opérations qu'effectuent les moyens de production sur le produit) [DBF98].

Un FMS se caractérise par :

- son haut degré d'automatisation des machines et des systèmes de transport
- sa capacité flexible de fabrication
- son utilisation élevée d'équipement
- un délai d'exécution court
- un stock d'en-cours (WIP) réduit
- une réponse améliorée du marché
- une fabrication 'juste à temps' (Just-in-Time)
- une fabrication simplifiée
- une commande de gestion améliorée
- une haute qualité
- un marché amélioré

Néanmoins, les FMS nécessitent une grande gestion de support, de matériel, de palette et de produit

L'un des objectifs les plus importants des systèmes flexibles de production (FMS) est de minimiser tous les facteurs provoquant des surcoûts sans donner de valeurs ajoutées aux produits. Parmi ces facteurs, il y a le stockage. Durant le stockage, le produit n'acquiert aucune valeur ajoutée. En contrepartie, il consomme du temps, de l'argent et de l'espace. De ce fait, les stocks sont maintenus à niveau bas, ce qui mène à de plus petits systèmes de stockage (minimiser l'investissement initial, le nombre de produits stockés, les durées de stockage...); cela crée en conséquence, des besoins d'accès rapide aux matériaux stockés, pour réduire au minimum les temps d'attente au niveau de la production.

Aussi, un FMS requiert-il un système de stockage pour entreposer ses charges. Ces systèmes de stockage doivent être conçus de telle sorte qu'ils peuvent fournir des charges de tous genres (outils, matériaux, palettes, produits) avec des temps de réponse rapide pour entretenir les demandes de fabrication, d'entreposage, et des applications de distribution ainsi qu'une utilisation minimale d'espace. Les systèmes automatisés de stockage/déstockage (Automated Storage/Retrieval Systems, AS/RS) sont une bonne alternative aux systèmes conventionnels de stockage pour atteindre les objectifs énoncés ci-dessus.

Un système automatisé de stockage déstockage est une combinaison d'équipements et de

systèmes de contrôle qui prend en charge, stocke et déstocke, des produits avec précision, exactitude et célérité sous un certain degré d'automatisation.

Un système automatisé de stockage/déstockage a pour fonction de recevoir, identifier, classer, mettre en stock, déstocker, emballer, des produits. Il sert aussi à emmagasiner les produits en cours de production, la matière première, les pièces détachées, les supports, les outils, les produits à retravailler

Depuis les débuts des années 50, les systèmes automatisés de stockage/déstockage sont très utilisés dans beaucoup de systèmes de production et de distribution, ce qui a mené au développement d'une grande variété de type d'AS/RS pour étendre tout le domaine de leur fonctionnement et pour être employés pour tout genre de produits. La classification des AS/RS est basée sur leurs structures et leurs applications, parmi les AS/RS les plus importants : les AS/RS à charge unitaire, à mini charge, à personne embarquée, à étagère profonde, à convoyeur gravitationnel, à rack glissant, multi allées, à racks glissants. L'AS/RS à charge unitaire représente le système générique. Les autres systèmes représentent des variations de ce système.

Les AS/RS ont une importance capitale dans la réussite des défis de la nouvelle industrie, pour tirer profit des meilleurs avantages de ces systèmes il faut bien les étudier ; malheureusement, mis à part certains types d'AS/RS (par exemple à charge unitaire), ils ne sont pas beaucoup étudiés, bien qu'ils soient très utilisés dans l'industrie et les services. Ce manque d'intérêt pour les AS/RS particuliers tels que l'AS/RS à convoyeur gravitationnel (Flow Rack Automated Storage/Retrieval System), fait qu'ils ne sont utilisés que pour des applications particulières et très en deçà de leurs capacités. Une étude approfondie permettra d'explorer tous les champs d'applications de tels systèmes.

L'AS/RS à convoyeur gravitationnel (Flow Rack Automated Storage/Retrieval System) est très peu étudié. Ce type d'AS/RS est utilisé dans l'industrie agro-alimentaire où un seul type de produits (ou très peu de produits différents) sont stockés. Les mêmes produits sont stockés dans les casiers et le déstockage se fait suivant la règle FIFO. Cette façon de l'utiliser limite considérablement son champ d'application.

Au sein de notre Laboratoire (MELT), plusieurs travaux ont concerné ce type d'AS/RS. Le but étant de faire une investigation des performances de l'AS/RS à convoyeur gravitationnel en stockage purement aléatoire. Dans le cas où ces performances sont intéressantes, le champ d'application de cet AS/RS s'élargira vers d'autres applications et pourra concurrencer l'AS/RS à charge unitaire, qui est actuellement le système le plus utilisé dans l'industrie et les services.

Sari *et al.* [SCG01], [ZS05] ont établi les expressions des temps de cycle des machines de stockage/déstockage, afin d'évaluer ses performances et de déterminer son comportement. Les résultats obtenus ont permis de montrer, qu'en termes d'espace utilisé par rapport à la capacité de stockage, l'AS/RS à convoyeur gravitationnel est beaucoup plus performant que l'AS/RS à charge unitaire. Par contre en termes de temps de cycle, l'AS/RS à convoyeur gravitationnel

présente des temps de cycle plus longs. Il est donc nécessaire d'améliorer ces temps de cycle pour pouvoir utiliser ce système dans les mêmes applications que l'AS/RS à charge unitaire.

Gaouar *et al.* [FEM04a], ont simulé une heuristique de stockage/déstockage développée par Sari *et al* afin de minimiser le temps de déstockage en vue d'améliorer les performances du système AS/R à convoyeur gravitationnel. Les résultats obtenus ont montré que le temps moyen de déstockage pour un système utilisant l'heuristique est nettement inférieur au temps moyen pour un stockage/déstockage purement aléatoire.

Pour notre part, l'objectif de notre travail est essentiellement de minimiser le temps de déstockage ; pour cela nous avons utilisé les « Méthaheuristiques » en les combinant avec l'heuristique de stockage/déstockage, développée par Sari *et al.* De ce fait, et dans ce travail, nous allons développer un simulateur, implémenter toutes les méthodes, étudier le comportement du système et analyser ses résultats.

La simulation, consiste à faire évoluer le modèle d'un système sur une période appropriée en fournissant les entrées adéquates, puis à observer et analyser les résultats, pour établir les meilleures politiques afin d'atteindre les objectifs attendus.

Le présent rapport est organisé en cinq chapitres. Le Chapitre I est une introduction aux systèmes automatisés de stockage/déstockage, où nous parlerons des différents aspects des AS/RS, de leur fonctionnalité, de leurs composants aux différents types d'AS/RS existants. Le Chapitre II est une description détaillée des systèmes AS/RS à convoyeur gravitationnel, de leur composants, de leur fonctionnement, ainsi que de la modélisation des temps de transport des machines de stockage et de déstockage de l'AS/RS à convoyeur gravitationnel ; enfin, il y sera fait une analyse des taux de fonctionnement des machines S/R du système AS/RS à convoyeur gravitationnel.

On introduira au Chapitre III, deux méthaheuristiques désormais classiques, la méthode Recuit Simulé et la méthode tabou, après avoir défini et classifié les méthaheuristiques. Dans le Chapitre IV nous présentons notre simulateur qui à été développé en langage de programmation java ; nous énumérerons les principales classes qui le composent.

Au Chapitre V, nous présenterons l'adaptation des deux méthaheuristiques(recuit simulé et recherche tabou), puis nous analyserons et interpréterons les résultats obtenus de la simulation ainsi que les conclusions de cette étude ; une étude comparative va ainsi être faite entre les performances des différentes méthodes de stockage/déstockage suivante : purement aléatoire, heuristique de stockage/déstockage, méthaheuristiques Recuit Simulé, méthaheuristiques Recherche Tabou, plus proche voisin, et les résultats du modèle analytique pour un système donné, par rapport au temps moyen de déstockage.

Chapitre I

Introduction aux Systèmes Automatisés de Stockage/Déstockage.

1 Introduction

L'automatisation industrielle a connu, au cours de ces dernières décennies, une évolution importante consécutive à l'accroissement des exigences de qualité, de flexibilité et de disponibilité dans les procédés industriels [JP04]. L'automatisation de ces derniers concerne tous les aspects de l'activité industrielle : production, assemblage, montage, contrôle, conditionnement, manutention, stockage, Son objectif est de réaliser, de manière automatique, des fonctions particulières répondant à des besoins spécifiques.

Les productions industrielles sont de plus en plus automatisées [SD00]. Cette progression du degré de l'automatisation concerne l'automatisation d'opérations autrefois manuelles comme les assemblages ou les contrôles, mais aussi l'automatisation plus poussée d'opérations déjà partiellement automatisées. Ceci se trouve, par exemple, dans le passage en automatique de machines semi-automatiques ou le remplacement de machines rigides (ne fabriquant qu'un seul type de produit) par des machines flexibles susceptibles d'opérer sur plusieurs variantes de produits.

Le stockage est parmi les systèmes automatisés qui sont nécessaires pour le bon fonctionnement des systèmes de production. Ce dernier doit être le plus petit possible (afin de minimiser l'investissement initial, le nombre de produits stockés, les durées de stockage). En même temps, il doit réagir très rapidement à une demande de stockage ou de déstockage, pour minimiser les temps d'attente au niveau de la production.

Un système automatisé de stockage/déstockage (Automated Storage/Retrieval system, AS/RS) est composé typiquement de racks de stockage, machines de stockage et déstockage, convoyeurs et de stations d'entrée/sortie où la machine de stockage/déstockage ;il peut charger et décharger des produits.

Il existe plusieurs types d'AS/RS tel que l'AS/RS à charge unitaire, à mini-charge, et l'AS/RS à convoyeur gravitationnel. Le type d'AS/RS le plus étudié est l'AS/RS à charge unitaire qui constitue le modèle générique des autres [KH08]. Dans ce chapitre nous présentons les définition, fonction, composition d'un système de stockage AS/RS à la section 2. Ensuite, à la section 3, nous définirons les différents types d'AS/RS. La section 4 sera consacrée à la conception des AS/RS et la section 5 aux différentes méthodologies de stockage. La section 6 traitera des différentes composantes des aires de stockage ainsi que leurs dispositions. Enfin à la section 7 un

état de l'art détaillé sur l'évaluation des performances des AS/RS sera présenté.

2 Système automatisé de stockage/déstockage AS/RS

2.1 Définition

Un système automatisé de stockage/déstockage (Automated Storage & Retrieval System, AS/RS) est défini par Materials Handling Institute [RDM77] comme étant une combinaison d'équipements et de systèmes de contrôle qui manipule, stocke, et déstocke des matériaux avec précision, exactitude et célérité, sous un certain degré d'automatisation.

2.2 Fonction

Un système automatisé de stockage/déstockage (AS/RS) a pour fonction de recevoir, identifier, classer, mettre en stock, déstocker, des produits. Il sert aussi à emmagasiner les produits en cours de production, la matière première, les pièces détachées, les palettes, les supports, les outils, les produits à retravailler [ZS05].

2.3 Composition

Un système automatisé de stockage/déstockage est composé [ZS05] :

- d'une série d'allées avec des étagères de stockage de part et d'autre,
- de machines de stockage/déstockage, en général une par allée,
- de points d'entrées/sorties,
- d'un système de contrôle orchestrant le tout.

2.4 Terminologie

- Casier : le casier est l'unité de stockage de base. C'est l'espace tridimensionnel nécessaire au stockage d'une charge unique ou d'un ensemble de charges.
- Baie : c'est la colonne formée de l'ensemble de casiers allant du sol au plafond.
- Rangée : c'est l'ensemble de casiers formés par des baies placées les unes à la suite des autres.
- Etagère : c'est la structure formant une rangée. C'est, en général, une structure d'acier, conçue pour supporter la totalité des charges placées dans chaque casier.
- Allée : l'allée est l'espace entre deux rangées où circule la machine de stockage/déstockage.
- Machine S/R : la machine de stockage/déstockage (Storage/Retrieval machine, S/R machine) est une structure mobile servant à stocker/déstocker les produits.

- Module de stockage : ce sont des palettes ou des containers de formes standardisées pouvant être placés dans les casiers, chargés par les machines S/R et déchargés dans les stations de dépôt/livraison.
- Station P/D : la station de dépôt/livraison (Pickup/Deposit station, P/D station), est aussi appelée station d'entrée/sortie. C'est le point par où transitent tous les produits entrant ou sortant du système. C'est aussi l'interface entre la machine S/R et le système de transfert externe à l'AS/RS.

3 Types d'AS/RS

Il existe différents types d'AS/RS. Leur classification est basée sur leurs structures et leurs applications. Nous distinguons :

- Les AS/RS à charge unitaire ;
- Les AS/RS multi allées ;
- Les AS/RS à racks glissants (ou racks mobiles) ;
- Les AS/RS à mini charge ;
- Les AS/RS à carrousel ;
- Les AS/RS à personne embarquée ;
- Les AS/RS à étagères profondes ;
- Les AS/RS à convoyeur gravitationnel.

3.1 Les AS/RS à charge unitaire

Les AS/RS à charge unitaire sont utilisés pour stocker des charges palettisées ou en container standard. Les charges sont d'ordinaire supérieures à 250 kg par unité. En général, le système est commandé par ordinateur. Il est équipé d'une machine de stockage/déstockage, appelée machine S/R, conçue pour traiter des containers ou palettes à charge unitaire. Chaque machine est placée sur des rails fixés au sol. La machine S/R est composée d'un cadre sur lequel coulisse, verticalement, un plateau, qui sert à transporter les charges. Ce plateau peut se déplacer horizontalement à la perpendiculaire du cadre. De ce fait, il peut charger/décharger les produits dans les casiers, ainsi qu'au niveau de la station de dépôt/livraison. Plusieurs types de mécanismes de support de la charge sont possibles [ZS05] :

- Le système à plateau coulissant : c'est le même principe que le transpalette.
- Le système à ventouse : la charge est accrochée en créant un vide dans des ventouses.
- Le système à aimant : approprié pour les produits en fer et acier.

Les machines S/R peuvent avoir des hauteurs de 6m à 40m et supportant des charges de 50kg à 5tonnes [N.S96].

Le système à charge unitaire est l'AS/RS générique. Les autres systèmes décrits ci-dessous représentent des variations de l'AS/RS à charge unitaire.

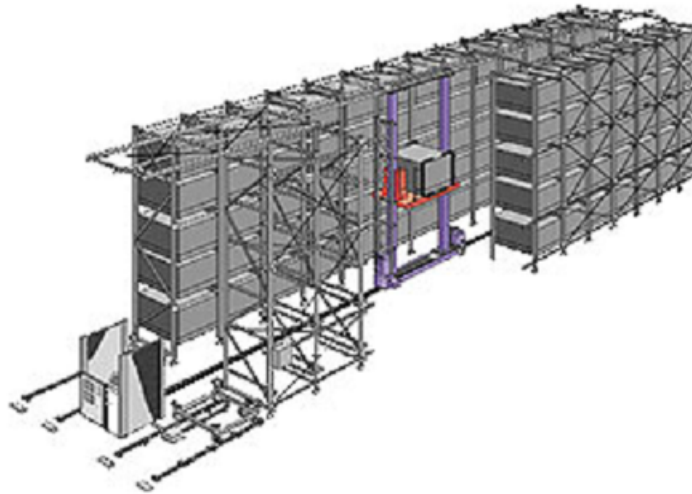


Figure I.1 – AS/RS à charge unitaire.

3.2 Les AS/RS multi allées

Un AS/RS multi allées est une variation des AS/RS à charge unitaire, composé d'un ensemble de racks disposés deux à deux et séparés par une allée appelée allée de desserte. Une allée commune placée perpendiculairement aux racks relie toutes les allées de desserte. Un tel système a une seule machine S/R capable de servir l'ensemble des racks. De ce fait, cette machine se déplace suivant trois axes : verticalement le long des colonnes formant les racks, horizontalement le long des allées de desserte et transversalement le long de l'allée commune dont l'une des extrémités est équipée d'une station de dépôt/livraison.

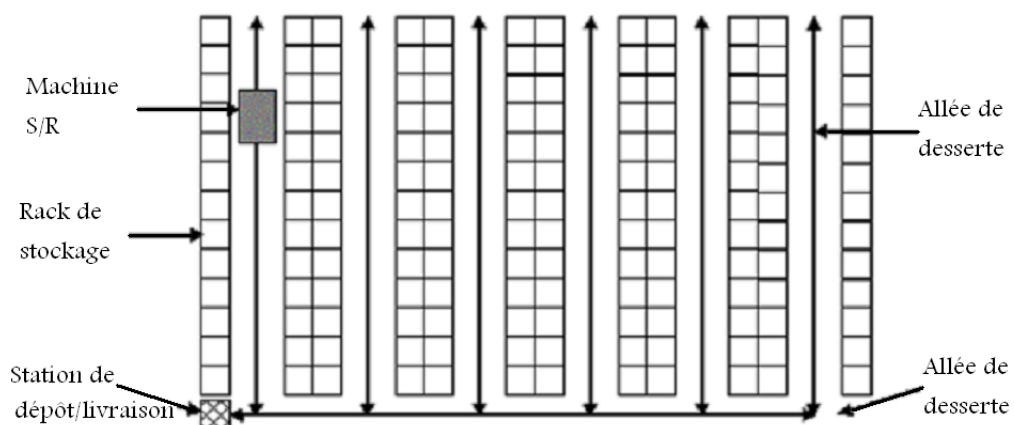


Figure I.2 – AS/RS multi allées.

3.3 Les AS/RS à racks glissants

Les AS/RS à racks glissants sont une variation des AS/RS multi-allées. La particularité de ces systèmes est que les allées de service n'apparaissent que lorsqu'une opération de stockage ou de déstockage est prévue dans les racks correspondants. Les racks formant le système glissent latéralement sur des rails de telle sorte que l'on puisse ouvrir une allée entre n'importe lequel des deux racks mitoyens. Au repos, il existe une seule allée de service, celle-ci pouvant se déplacer entre n'importe lequel des deux racks mitoyens ; ce n'est pas l'allée qui se déplace mais les racks [ZS05].

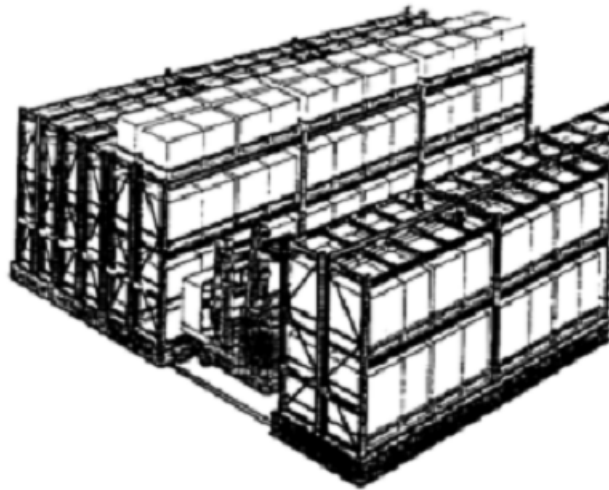


Figure I.3 – AS/RS à racks glissants.

3.4 Les AS/RS à mini charge

Les AS/RS à mini charge sont utilisés pour stocker de petites charges (pièces détachées, outils,...), qui sont contenues dans des containers, un ensemble de produits semblables ou non, sont stockés dans chaque containers. En cas de stockage/déstockage, c'est le container en entier qui est déplacé vers un opérateur qui va procéder à la sélection des produits à déstocker ou à stocker (l'opérateur peut être une personne ou un robot). Le système AS/R à mini charge est généralement plus petit que l'AS/RS à charge unitaire ; il est souvent enfermé pour la sécurité des articles stockés et nécessite un investissement relativement faible.

3.5 Les AS/RS à carrousel

Les carrousels sont une bonne alternative au systèmes à mini charge [JT84]. Les AS/RS à carrousels sont constitués de bacs rotatifs pourvus de tablettes. Ces bacs glissent sur des rails

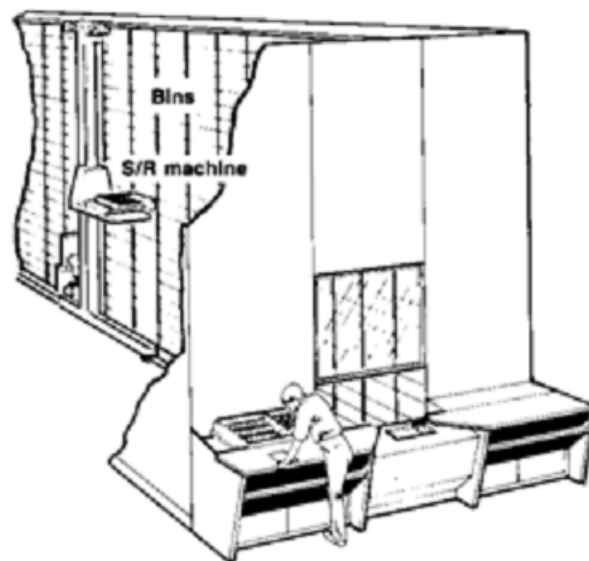


Figure I.4 – AS/RS à mini charge

formant un circuit oval. Le mouvement du carrousel permet de livrer les articles à l'opérateur. Celui-ci peut ainsi prélever un article sur un carrousel en action pendant que les autres se mettent en position pour lui permettre de poursuivre sa cueillette en se tournant vers eux dès qu'il est prêt à le faire. On trouve deux types : carrousel horizontal et carrousel vertical.



Figure I.5 – AS/RS à carrousel horizontal

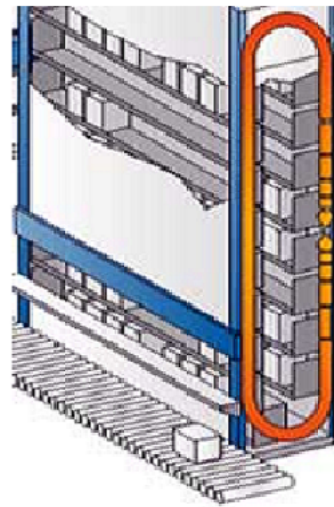


Figure I.6 – AS/RS à carrousel vertical

3.6 Les AS/RS à personne embarquée

Les AS/RS à personne embarquée sont utilisés pour stocker plusieurs petits produits dans chaque casier. Une personne montée sur une plate-forme mobile sélectionne les différents produits à déstocker directement de leurs endroits de stockage ou les stocke dans leur casier adéquat. Ce système permet de stocker/déstocker plusieurs produits en une seule opération. Ce qui permet de réduire les temps de stockage/déstockage. L'opérateur, après les avoir déstockés, charge les produits sur la machine S/R qui les transporte vers la station de dépôt/livraison. La plate-forme, sur laquelle se trouve l'opérateur, peut contenir des systèmes automatiques de levage, pour charger/décharger les produits lourds.

3.7 Les AS/RS à personne embarquée

Les AS/RS à étagères profondes sont une variation de l'AS/RS à charge unitaire. Les produits peuvent être stockés dans le même casier (jusqu'à 10 produits) l'un après l'autre, et donc, possédant la même adresse. Cela amène à une haute densité de stockage, autorisant un haut usage de l'espace.

Au niveau de chaque casier, les produits peuvent se déplacer horizontalement. Ils sont stockés d'un côté du casier et déstockés de l'autre côté. Ce déplacement est possible grâce à un plateau mobile qui se déplace de la machine S/R à l'entrée du casier. La machine S/R est identique à celle d'un AS/RS à charge unitaire avec en plus des fonctions spécifiques pour lui permettre le contrôle des plateaux mobiles.

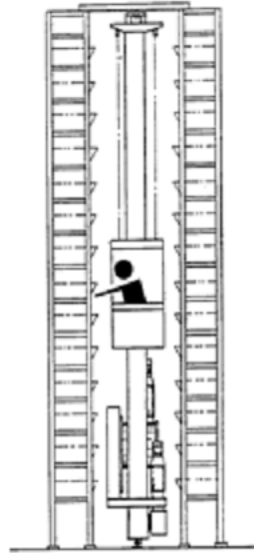


Figure I.7 – AS/RS à personne embarquée

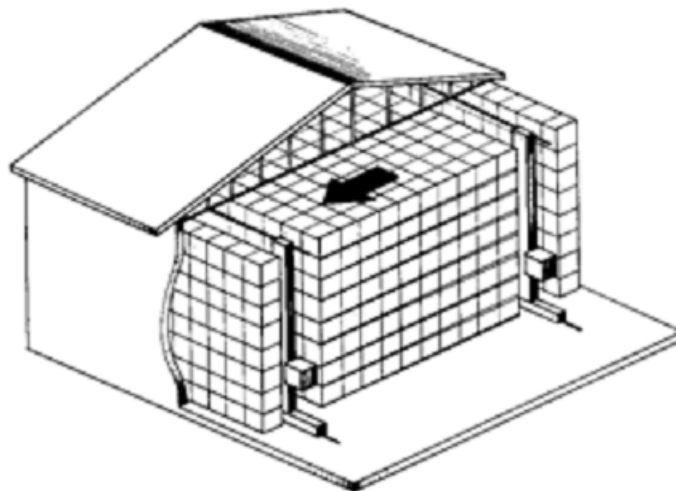


Figure I.8 – AS/RS à étagères profondes

3.8 Les AS/RS à convoyeur gravitationnel

Ce système est une variation des AS/RS à étagère profondes. Il est conçu pour stocker soit des produits individuels, soit des charges en palettes ou container. Il est constitué d'un ensemble de casiers profonds dont les supports sont des convoyeurs gravitationnels. Deux machines S/R se trouvent aux deux extrémités, une sur une face assure le stockage et l'autre sur l'autre face assure le déstockage, les deux machines sont reliées entre elles par un convoyeur de restockage incliné en sens inverse permettant à la machine de stockage d'accéder aux produits pour les

restocker. Une description plus détaillée de ce système sera présentée au chapitre 2.

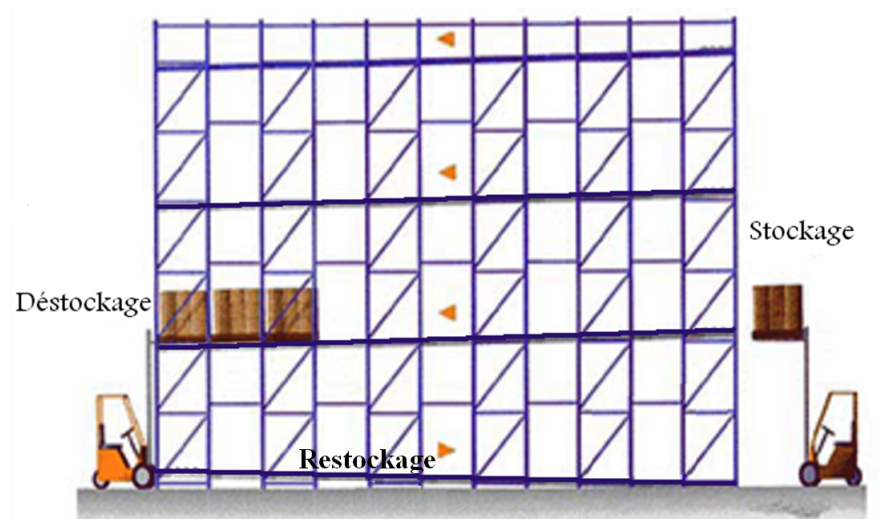


Figure I.9 – AS/RS à convoyeur gravitationnel

4 Conception d'un AS/RS

Le Material Handling System Institute [RDM77] ainsi que Bozer et White [YB80] dressent une liste de paramètres qui doivent être pris en considération pour une bonne conception d'un système AS/RS. Parmi les plus importants paramètres, nous citons :

- Dimensions de la charge : largeur, hauteur, profondeur, poids,
- Dimensions du casier : largeur, hauteur, profondeur,
- Nombre total de casiers
- Structure de l'étagère : charge utile, charge admissible,...
- Rendement : le nombre de stockage/déstockage par unité de temps
- Durée de cycle : cycle simple, cycle double, cycle multi-adresses,...
- Taux d'utilisation des machines.

En concevant un AS/RS, deux principes fondamentaux doivent être considérés [JW82]. Le premier est que l'AS/RS doit fonctionner, au maximum, à 85% de ses capacités maximales. Le second est qu'il faut considérer chaque cas de conception d'AS/RS comme un problème unique.

5 Méthodologies de stockage

La méthodologie de stockage spécifie l'emplacement du produit dans les zones de stockage selon des règles imposées. On distingue trois modes pour les systèmes de stockage :

- Le stockage dédié où chaque type de produit a son propre secteur de stockage.
- Le stockage aléatoire ouvert où chaque produit est assigné à un casier de manière aléatoire.
- le stockage aléatoire par classe où les produits sont divisés généralement en trois classes suivant leur temps et fréquence de stockage [Sar97].

5.1 Stockage dédié

Dans le stockage dédié, chaque type de produit a une zone de stockage déterminé. Ce qui fait que l'aire de stockage global est divisée en petits blocs dédiés pour chaque type de produit. Le contrôle de ce système et la gestion des stocks sont simplifiés, Néanmoins, l'espace de stockage augmente puisqu'il faut donner suffisamment d'espace pour les stocks maximums de chaque produit et le taux de remplissage du système de stockage est relativement bas [ZS05]. L'ajout de nouveaux produits non prévus à l'origine est très complexe.

5.2 Stockage aléatoire ouvert

Dans le stockage aléatoire, n'importe quel produit peut être stocké dans n'importe quel casier. Un système de contrôle élaboré garde en mémoire les adresses de chaque type de produit. Contrairement au stockage dédié, le stockage aléatoire utilise beaucoup moins d'espace. L'espace total de stockage devant être équivalent au stock maximum de tous les produits. Ceci permet une réduction importante d'espace. Askin et standridge [Ask93] présente un exemple de stockage aléatoire où le gain d'espace, par rapport au stockage dédié, est de l'ordre de 40%. Cependant, les systèmes de stockages aléatoires nécessitent un contrôle très élaboré.

5.3 Stockage aléatoire par classe

Le stockage aléatoire permet de réduire l'espace de stockage par rapport au stockage dédié. Le stockage par classe s'attaque au temps de cycle. Des réductions du temps de cycle peuvent atteindre 40% [Ask93]. Les produits dans un AS/RS ne restent pas stockés dans la même durée. Si les produits qui restent stockés peu de temps sont placés près de la station de livraison et les produits qui restent plus longtemps sont placés loin de la station, le rendement du système pourra être grandement amélioré. Le stockage par classe subdivise les casiers en classes par rapport à leurs distances du point de livraison, les casiers les plus proches formant la première classe, les plus loin formant la dernière. Les produits à stocker sont aussi classés suivant le temps qu'ils vont passer dans le système. Ceux qui restent le moins longtemps seront stockés

dans la première classe et ceux qui restent le plus longtemps dans la dernière [ZS05] .

6 Disposition de l'aire de stockage

La disposition du système de stockage doit avoir comme objectifs de minimiser les coûts d'investissement initiaux et le temps de stockage/déstockage. Une disposition mal conçue entraînerait de grandes pertes à l'investissement et au fonctionnement.

6.1 Le Bâtiment

Les dimensions et la forme du bâtiment dépendent du type de système AS/RS qu'il doit abriter. Pour que l'espace soit utilisé au maximum, une forme parallélépipédique paraît être la plus appropriée pour un AS/RS classique.

6.2 Les Allées

La longueur et la position des allées influent sur le temps de réponse du système de stockage. Des allées plus longues nécessitent des temps moyens de stockage/déstockage plus importants que des allées plus courtes.

6.3 Les étagères

La hauteur, la profondeur, le type d'étagères influent sur les performances globales du système. Il existe plusieurs types d'étagères :

- Les étagères à charge multiples : les casiers ont une profondeur telle qu'ils peuvent stocker plusieurs produits de même type ; la règle de stockage sera LIFO (dernier entré, premier sorti)
- Les étagères pour mini et micro charges
- Les étagères à convoyeur gravitationnel : chaque casier d'une profondeur importante est pourvu d'un convoyeur gravitationnel. Le casier est accessible, des deux cotés, par deux machines S/R : l'une pour le stockage, l'autre pour le déstockage. La règle de stockage/déstockage est FIFO (premier entré, premier sorti).

7 L'évaluation des performances des AS/RS dans la littérature

L'amélioration des performances des différents types d'AS/RS a suscité l'intérêt de beaucoup de chercheurs. Le temps de cycle des machines de stockage/déstockage (S/R), qui a été l'objet de plusieurs travaux, est l'un des plus importants critères de performance. Hausman *et al.* [WHH00] ont été parmi les premiers à développer un modèle analytique pour l'étude des AS/RS à charge unitaire. Ils ont abordé le problème de l'assignement optimal de stockage. Ils ont considéré trois stratégies : le stockage aléatoire, le stockage dédié et le stockage par classe. Ils ont montré qu'une réduction importante du temps de cycle est obtenue dans le cas d'un stockage dédié en comparaison avec un stockage purement aléatoire. Bozer et White, [YB84], ont développé des expressions mathématiques permettant de calculer le temps de simple et de double cycle pour un rack à charge unitaire rectangulaire. Ils ont utilisé une approximation continue du rack avec une politique de stockage aléatoire en explorant plusieurs configurations du point d'entrée/sortie et de point de repos de la machine S/R. Dans Han *et al.* [MHH87], les auteurs développent une expression mathématique estimant le temps du double cycle, sous la règle « plus proche voisin » (nearest-neighbour). Sous cette règle, les requêtes de stockage et les requêtes de déstockage les plus proches sont appariées pour minimiser le temps de double cycle. Hwang *et al.* [HH00] ont proposé une expression mathématique pour les AS/RS multi allées ; dans ce travail, chaque rack est considéré comme une face continue et pour chaque rack, ils ont développé une expression analytique du temps de cycle. Ghomri *et al.* [LG08] ont proposé une expression analytique du temps moyen de simple cycle d'un AS/RS multi allées en considérant que l'ensemble des racks représentent un espace continu.

D'autres auteurs ont étudié le problème de la position du point de repos de la machine S/R et son impact sur le temps de cycle. Egbelu [Egb91] a développé des formulations pour minimiser le temps de cycle et les temps de réponse maximum. Egbelu et Wu [EW93] ont utilisé la simulation pour comparer les règles de positionnement du point de repos. Hwang et Lim [HH93] utilisèrent les règles développées par Egbelu [Egb91] et les transformèrent en un problème de positionnement unique en utilisant les distances Tchebyshev, pour la première règle, et le minimax de Tchebyshev, pour la seconde. Peters *et al.* [BAPH96] ont développé une expression analytique de la position du point de repos sous une variété de configurations d'AS/RS concluant que le meilleur point de repos de la machine S/R se trouve au milieu du rack. Chang et Egbelu [CE97a] et [CE97b] ont développé deux algorithmes pour la détermination du point de repos optimal d'une machine S/R desservant un AS/RS multi allées. Dans Meller *et al.* [RDM05], les auteurs ont fait une investigation par simulation sur l'impact du point de repos sur le temps moyen de transport et ils ont trouvé que le choix du point de repos de la machine S/R sous un taux élevé d'utilisation n'affecte pas significativement le temps de réponse du système. En s'inspirant du travail de Bozer *et al.* précédent, Sari *et al.* [ZS05] ont développé deux modèles (approché et exact) du temps de transport pour l'AS/RS à convoyeur gravitationnel. Le modèle approché est basé sur une approche continue, où le rack est supposé continu. Le modèle exact est basé sur une approche discrète. Les expressions des deux approches sont comparées via la simulation. Les auteurs concluent qu'il n'y a pas de différence significative entre les résultats obtenus par

les deux approches mais l'approche continue est plus pratique, à cause de sa simplicité de calcul par rapport à l'approche discrète qui est extrêmement complexe.

Un autre axe de recherche dans les AS/RS concerne l'ordonnancement des opérations de stockage/déstockage dans le but d'améliorer le temps de réponse du système. Lee [Lee92] considéra le problème d'attribution d'espace de stockage dédié de sorte que le temps total de cycle est minimisé dans un AS/RS à personne à bord. Une heuristique pour la disposition des espaces de stockage est développée basée sur la technologie de groupe. Dans [Lee96], les auteurs ont étudié un problème d'ordonnancement des demandes de déstockage dans un AS/RS à charge unitaire. Dans le but de réduire le temps total de transport de la machine S/R, ils ont introduit une méthode qui trouve une solution optimale ou quasi-optimale pour des problèmes de taille moyenne. Dans une étude de [BIK03], les auteurs proposent un nouveau processus de remplissage de racks interchangeables en fonction des demandes de clients. L'objectif est de minimiser les temps de préparation et de lancement des produits demandés. Gaouar [FEM04a] a proposé une heuristique pour améliorer les performances d'un AS/RS à convoyeur gravitationnel. Dans [JPVdb99], pour réduire le temps de réponse du système, les auteurs se sont intéressés à l'ordonnancement des opérations de stockage/déstockage en utilisant l'approche d'ordonnancement par bloc avec un stockage dédié. L'objectif était de trouver un chemin avec un temps total de transport minimum. Le problème est équivalent au problème du voyageur de commerce et ils ont montré que dans le cas particulier avec une politique de stockage dédié, le problème peut être résolu en un temps polynomial. Les auteurs dans [FA05] développent deux algorithmes de commande pour un AS/RS modélisé par des réseaux de Petri colorés temporisés. Ils ont introduit un système d'optimisation en temps réel, dans l'architecture de contrôle, afin d'améliorer le débit du système. [MD05] ont proposé une modélisation modulaire par les réseaux de Petri colorés temporisés d'un AS/RS à charge unitaire multi-allées, servi par des véhicules guidés par rails.

8 Conclusion

Dans ce chapitre, nous avons présenté les systèmes automatisés de stockage/déstockage ; de façon générale, nous avons énuméré quelques types, tels que l'AS/RS à charge unitaire, à mini-charge, et l'AS/RS à convoyeur gravitationnel... ; nous avons aussi parlé de leur conception et de leur utilisation. Ensuite nous avons présenté un état de l'art des recherches sur l'évaluation des performances des AS/RS. Dans cette partie, nous remarquons le grand intérêt que porte la communauté scientifique aux systèmes automatisés de stockage et surtout aux AS/RS à charge unitaire ; néanmoins, il y a des AS/RS qui ne sont pas beaucoup étudiés dans la littérature tels que les AS/RS à convoyeur gravitationnel ; de ce fait, nous nous sommes intéressés à ce système auquel nous avons consacré notre étude.

Chapitre II

Systèmes Automatisés de Stockage/Déstockage à Convoyeur Gravitationnel

1 Introduction

Un système automatisé de stockage/déstockage (Automated Storage/Retrieval system, AS/RS) est une combinaison d'équipements et de système de contrôle qui prend en charge, stocke et déstocke des produits avec précision, exactitude et célérité sous un certain degré d'automatisation [RDM77]. Un AS/RS est composé de racks de stockage, machines de stockage et déstockage, convoyeurs et de stations d'entrée/sortie où la machine de stockage/déstockage peut charger et décharger des produits. Il existe plusieurs types d'AS/RS tel que l'AS/RS à charge unitaire, à mini-charge, et l'AS/RS à convoyeur gravitationnel.

Ce chapitre est composé de deux parties principales : dans la première partie nous présentons les composantes, le fonctionnement d'un système AS/RS à convoyeur gravitationnel, ainsi que les expressions analytiques de ses différentes configurations ; la seconde partie est réservée à l'heuristique de stockage/déstockage pour l'AS/RS à convoyeur gravitationnel développée par Sari [ZS05] et simuler par Gaouar *et al* [FEM04a].

2 Système AS/RS à Convoyeur Gravitationnel

Notation : Avant d'entamer notre étude, nous allons définir les différents termes que nous utiliserons :

$\overline{E(RC)}$	temps moyen de déstockage ;
$\overline{E(V)}$	temps moyen de transport en deux points de la face du rack ;
L, H, D	longueur, hauteur et profondeur du rack AS/RS à CG ;

II.2 Système AS/RS à Convoyeur Gravitationnel

l, h, d	longueur, hauteur et profondeur d'un emplacement de stockage ;
M	nombre d'emplacements dans un casier, nombre de couches dans un rack ;
m	numéro de couche où se trouve le produit désiré ;
N_l	nombre de casier pour chaque ligne ;
N_h	nombre de casier pour chaque colonne ;
t_h	temps de transport horizontal de la station de dépôt/livraison à la colonne la plus éloignée ;
t'_h	temps de transport horizontal entre deux casiers mitoyen ;
t_v	temps de transport vertical de la station de dépôt/livraison à la ligne la plus éloignée ;
t'_v	temps de transport vertical entre deux casiers mitoyen ;
b	facteur de forme avec $b = \min \left(\frac{t_h}{T} ; \frac{t_v}{T} \right)$, ce qui implique que $0 < b \leq 1$
T	facteur de normalisation avec $T = \max (t_h, t_v)$;
ρ	taux de charge.

2.1 Composantes

L'AS/RS à convoyeur gravitationnel est composé de :

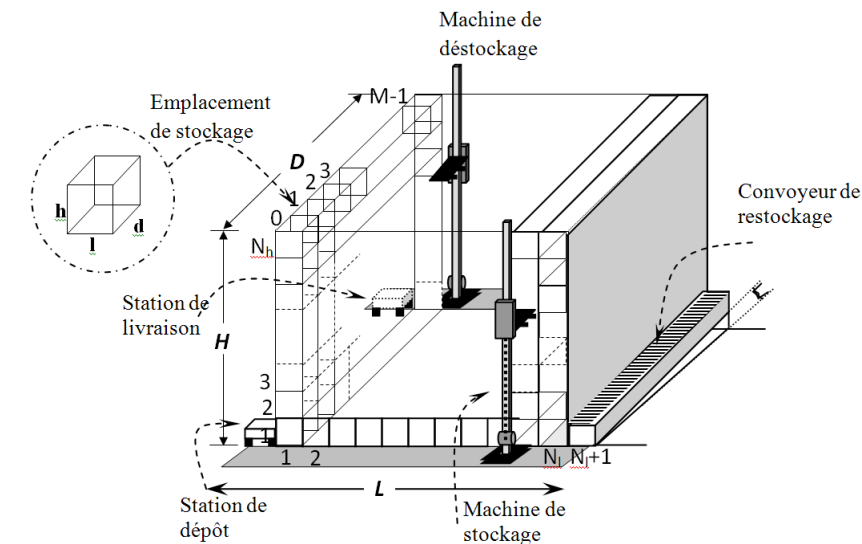


Figure II.1 – Configuration AS/RS à Convoyeur Gravitationnel

1. Un rack composé de casiers, d'une hauteur H , largeur L et profondeur D . il contient N_l casiers horizontaux et N_h casiers verticaux. Chaque casier est formé de M emplacements

II.2 Système AS/RS à Convoyeur Gravitationnel

de stockage numéroté de 0 à M-1, chaque emplacement a une largeur $l = L/N_l$, une hauteur de $h = \frac{H}{N_h}$ et une profondeur de $d = \frac{D}{M}$. les emplacements ayant le même numéro forme une couche. Chaque casier est muni d'un convoyeur gravitationnel à base de rouleaux ou roues libres inclinées de façon à permettre le glissement des produits d'un bout à l'autre du casier, donc d'une face à l'autre du rack.

2. Une machine de stockage sur la face avant du rack, et une machine de déstockage sur l'autre face. Les deux machines peuvent se déplacer simultanément sur deux axes. Ces deux axes forment le plan x-y parallèle aux deux faces du rack. elles possèdent la même vitesse horizontale V_h et la même vitesse verticale V_v . par conséquent, les temps de transport peuvent être calculés comme suit :
 - Le temps de déplacement horizontal de la station de dépôt/livraison à la colonne la plus éloignée : $t_h = \frac{L}{V_h}$;
 - Le temps de déplacement vertical de la station de dépôt/livraison à la ligne la plus éloignée : $t_v = \frac{H}{V_v}$;
 - Le temps de déplacement horizontal d'un casier au suivant : $t'_h = \frac{l}{V_h} = \frac{t_h}{N_l}$ avec $l = \frac{L}{N_l}$
 - Le temps de déplacement vertical d'un casier au suivant : $t'_v = \frac{h}{V_v} = \frac{t_v}{N_h}$ avec $h = \frac{H}{N_h}$
3. Une station de dépôt est située sur la face de stockage, où la machine de stockage récupère les produits à stocker. Une station de livraison est située sur la face de déstockage, où la machine de déstockage dépose les produits pour la livraison. Elles sont situées respectivement au coin bas de la face de stockage et de la face de déstockage.
4. Un convoyeur de restockage qui est un convoyeur gravitationnel, incliné au sens inverse, reliant les deux faces du rack, Il permet de transférer les produits à restocker vers la station de dépôt.

2.2 Fonctionnement du système AS/R à Convoyeur Gravitationnel

Pour stocker un produit, la machine, le récupère au point de dépôt et le dépose dans un casier de la face avant du rack. Il glisse sur les convoyeurs gravitationnels et se positionne dans l'emplacement libre le plus proche de la face arrière du rack. Pour déstocker un produit qui se trouve en position $m(m < M)$, la machine de déstockage déstocke tous les produits qui le précèdent un par un et les dépose au convoyeur de restockage où ils seront transférés vers la

station de dépôt pour un restockage.

2.3 Modélisation des temps de transport d'un AS/RS à Convoyeur Gravitationnel

Bozer et White [YB84] ont développé des expressions analytiques du temps de simple cycle dans les AS/RS à charge unitaire. Pour ce faire, ils ont utilisé une approche statistique basée principalement sur l'approximation des distances des casiers du rack en fonction continue. Leur étude est construite sur les hypothèses suivantes :

1. La face du rack est considérée comme étant une surface continue où les produits peuvent être stockés/déstockés en n'importe quel point.
2. La station de dépôt/livraison est située au coin bas, à l'extrémité gauche du rack.
3. La machine S/R se déplace simultanément dans les directions horizontale et verticale (Tchebychev Travel), avec des vitesses de déplacement constantes et connues.
4. Le stockage aléatoire est utilisé; cela veut dire que n'importe quel point de la face du rack a la même probabilité de subir une opération de stockage/déstockage.
5. Les temps associés au chargement/déchargement du produit par la machine sont ignorés. Ces temps sont, généralement, indépendants de la forme du rack et des vitesses de la machine S/R. ainsi, ils peuvent être ajoutés, à posteriori, aux expressions du temps de cycle.

Pour développer des modèles analytiques représentant les temps de stockage et de déstockage d'un système automatisé de stockage/déstockage à convoyeur gravitationnel, Sari *et al* [ZS05] ont basé leur étude sur les travaux de Bozer et White [YB84] sur l'AS/RS à charge unitaire tout en gardons les mêmes hypothèses.

Ainsi, les expressions du temps moyen sont données :

- Pour l'opération de stockage [YB84] :

$$\overline{E(SC)} = T\left(\frac{b^2}{3} + 1\right) \quad (\text{II.1})$$

II.2 Système AS/RS à Convoyeur Gravitationnel

- Pour l'opération de déstockage [ZS05] :

$$\overline{E(RC)}_m = \overline{E(V_1)} + \overline{E(V_2)} + m \cdot \overline{E(V_3)} \quad (\text{II.2})$$

avec :

- $\overline{E(V_1)}$:représente le temps moyen de transport entre le point de repos et le point de déstockage ;
- $\overline{E(V_2)}$:représente le temps moyen de transport entre le point de déstockage et la station de livraison ;
- $\overline{E(V_3)}$:représente le temps moyen pour retirer tous les produits qui se trouvent avant le produit à déstocker plus le temps pour remettre ces produits de l'autre coté.

Les expressions $\overline{E(V_1)}$, $\overline{E(V_2)}$ et $\overline{E(V_3)}$ peuvent être obtenues en se basant sur les travaux de Bozer et White [YB84] et de Peters *et al.* [BAPH96] et Sari *et al.* [ZS05] comme :

$$\overline{E(V_1)} = \frac{T}{4} \left(\frac{b^2}{3} + 1 \right) \quad (\text{II.3})$$

$$\overline{E(V_2)} = \frac{T}{2} \left(\frac{b^2}{3} + 1 \right) \quad (\text{II.4})$$

$$\overline{E(V_3)} = T \left(\frac{b^2}{3} + 1 \right) \quad (\text{II.5})$$

L'équation II.2 devient en remplaçant ses termes par leurs valeurs respectives définies dans les équations II.3, II.4 et II.5 :

$$\overline{E(RC)}_m = \frac{3}{4} T \left(\frac{b^2}{3} + 1 \right) + m \cdot T \left(\frac{b^2}{3} + 1 \right) \quad (\text{II.6})$$

avec

$$m = \rho M - 1 \quad 0 \leq m \leq M - 1$$

Finalement, l'expression du temps moyen de déstockage de l'AS/RS à convoyeur gravitationnel est :

$$\overline{E(RC)}_m = \frac{3}{4} T \left(\frac{b^2}{3} + 1 \right) \quad \text{pour } \rho < \frac{1}{M} \quad (\text{II.7})$$

II.2 Système AS/RS à Convoyeur Gravitationnel

$$\overline{E(RC)}_m = T\left(\frac{b^2}{3} + 1\right)\left(\frac{1}{4} + \frac{1}{2}\rho.M\right) \quad \text{pour } \frac{1}{M} \leq \rho \leq 1 \quad (\text{II.8})$$

avec :

$$t_h = t'_h.N_l$$

$$t_v = t'_v.N_h$$

$$T = \max(t_h, t_v)$$

$$b = \min\left(\frac{t_h}{T}, \frac{t_v}{T}\right) \quad 0 < b \leq 1$$

Sari *et al* [SCG01] [ZS05] et [ZS03] ont remarqué des variations importantes dans l'expression du temps moyen de déstockage lorsqu'il changé la position des stations de dépôt/livraison, ainsi que le convoyeur de restockage. De ce fait, ils ont développé les expressions analytiques des temps moyens de stockage et de déstockage pour chacune des configurations du système AS/R à convoyeur gravitationnel. Nous les présentons ci-dessous :

Cas A : La station de dépôt/livraison et le convoyeur de restockage aux extrémités opposées du rack.

$$\overline{E(SC)} = T\left(\frac{b^2}{3} + 1\right) \quad (\text{II.9})$$

$$\overline{E(RC)}_m = \frac{3}{4}T\left(\frac{b^2}{3} + 1\right) \quad \text{pour } \rho < \frac{1}{M} \quad (\text{II.10})$$

$$\overline{E(RC)}_m = T\left(\frac{b^2}{3} + 1\right)\left(\frac{1}{4} + \frac{1}{2}\rho.M\right) \quad \text{pour } \frac{1}{M} \leq \rho \leq 1 \quad (\text{II.11})$$

Cas B : La station de dépôt/livraison à l'extrémité du rack et le convoyeur de restockage au centre du rack.

$$\overline{E(SC)} = T\left(\frac{b^2}{3} + 1\right) \quad (\text{II.12})$$

$$\overline{E(RC)}_m = \frac{3}{4}T\left(\frac{b^2}{3} + 1\right) \quad \text{pour } \rho < \frac{1}{M} \quad (\text{II.13})$$

$$\overline{E(RC)}_m = T\left(\frac{b^2}{3} + 1\right)\left(\frac{1}{2} + \frac{1}{4}\rho.M\right) \quad \text{pour } \frac{1}{M} \leq \rho \leq 1 \quad (\text{II.14})$$

Cas C : La station de dépôt/livraison au centre du rack et le convoyeur de restockage à l'extrémité du rack.

II.2 Système AS/RS à Convoyeur Gravitationnel

$$\overline{E(SC)} = \frac{T}{2} \left(\frac{b^2}{3} + 1 \right) \quad (\text{II.15})$$

$$\overline{E(RC)}_m = \frac{T}{2} \left(\frac{b^2}{3} + 1 \right) \quad \text{pour } \rho < \frac{1}{M} \quad (\text{II.16})$$

$$\overline{E(RC)}_m = T \left(\frac{b^2}{3} + 1 \right) \left(\frac{1}{2} \rho \cdot M \right) \quad \text{pour } \frac{1}{M} \leq \rho \leq 1 \quad (\text{II.17})$$

Cas D : La station de dépôt/livraison et le convoyeur de restockage au centre du rack.

$$\overline{E(SC)} = \frac{T}{2} \left(\frac{b^2}{3} + 1 \right) \quad (\text{II.18})$$

$$\overline{E(RC)}_m = \frac{T}{2} \left(\frac{b^2}{3} + 1 \right) \quad \text{pour } \rho < \frac{1}{M} \quad (\text{II.19})$$

$$\overline{E(RC)}_m = T \left(\frac{b^2}{3} + 1 \right) \left(\frac{1}{4} + \frac{1}{4} \rho \cdot M \right) \quad \text{pour } \frac{1}{M} \leq \rho \leq 1 \quad (\text{II.20})$$

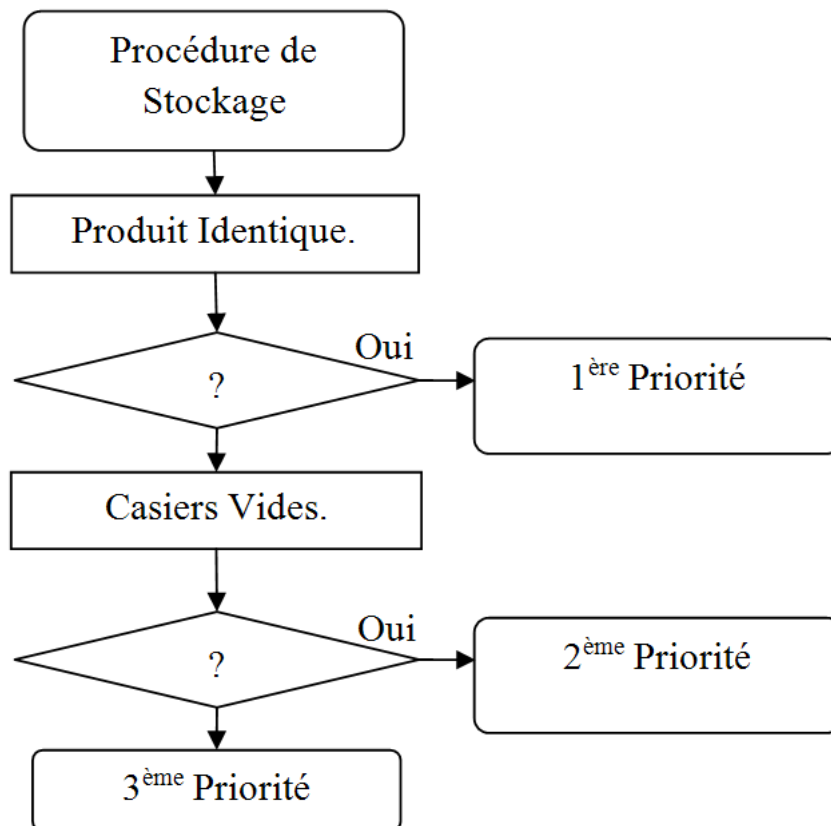
Après une analyse de l'impact des positions des stations dépôt/livraison ainsi que celles du convoyeur de restockage sur les temps moyens de stockage et déstockage de l'AS/RS à convoyeur gravitationnel et l'étude des différents modèles développés pour chacune des positions. Sari [ZS05] a déduit que :

1. la meilleur configuration est lorsque le convoyeur de restockage est au centre du rack sauf pour de petites valeurs de M et ρ .
2. La position de la station de livraison a un faible effet sur le temps de déstockage sauf pour de petites valeurs de M et ρ .
3. Lorsque M ou ρ augmente, le temps moyen de déstockage par unité, augmente lorsque le convoyeur de restockage est au coin du rack et diminue lorsqu'il est au centre du rack.
4. Le meilleur système est lorsque la station de livraison et le convoyeur de restockage sont au centre du rack.

3 Heuristique pour le stockage/déstockage dans un AS/RS à convoyeur gravitationnel

Dans cette section, nous présentons une heuristique de stockage/déstockage pour l'AS/RS à convoyeur gravitationnel [ZS05]. Sari a construit cette heuristique de manière intuitive pour essayer de réduire au maximum le temps moyen de déstockage. La réduction du temps de stockage (celui-ci étant très petit par rapport au temps de déstockage) n'a pas été prise en considération dans le développement de cette heuristique.

L'heuristique de stockage/déstockage est constituée de deux procédures. La première procédure est pour le stockage ; elle consiste en trois priorités à suivre ; si la condition de la première n'est pas valable, on passe à la seconde sinon on passe à la troisième. Ceci est représenté dans l'organigramme 2.1. La deuxième procédure est pour le déstockage ; elle consiste à chercher le produit désiré se trouvant dans la couche la plus proche de la machine de déstockage.



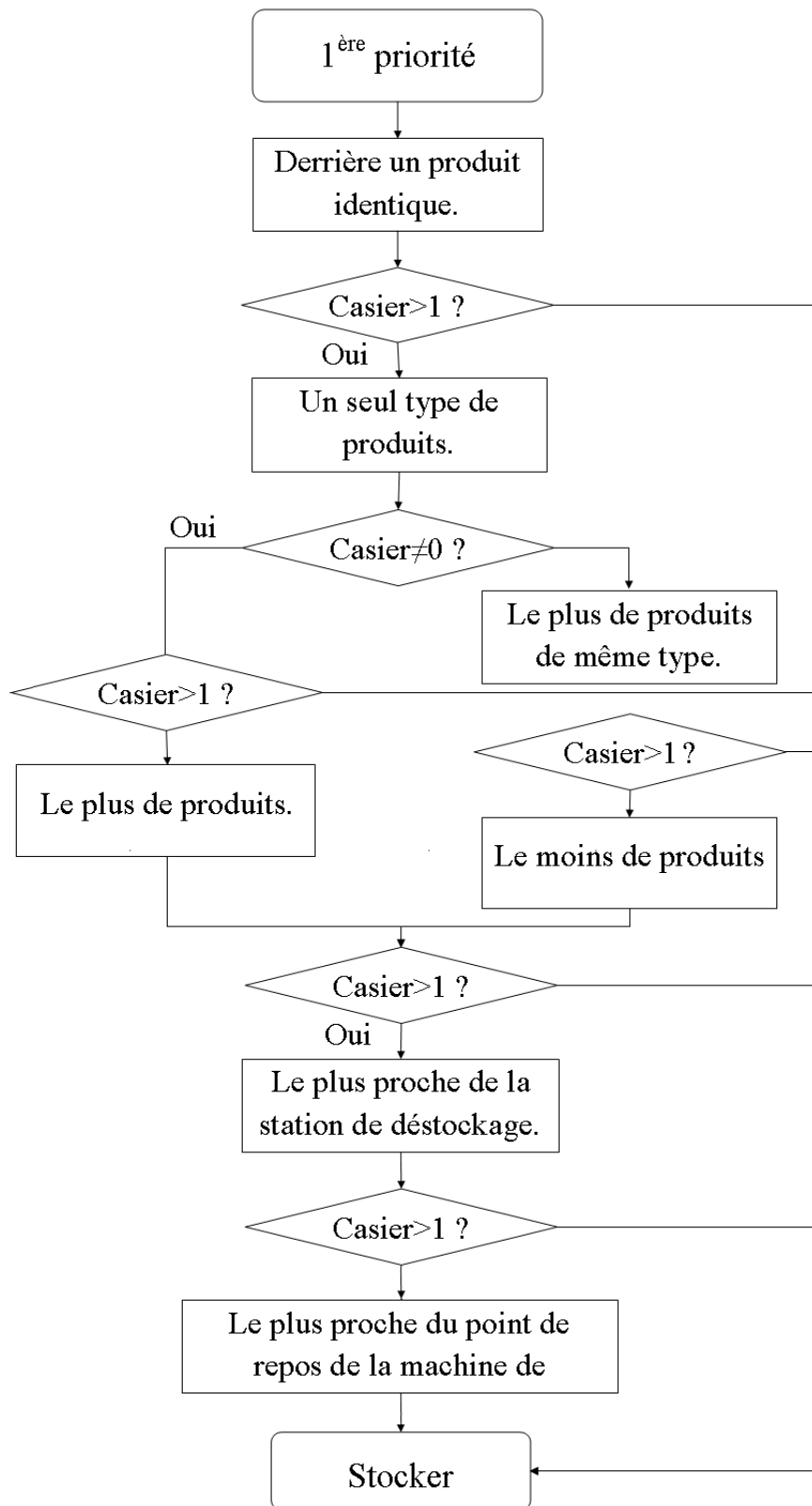
Organigramme 2.1

3.1 Procédure de stockage

1. Stocker le produit en cours derrière un produit identique Si un seul casier, Stocker Si plusieurs casiers,
 - (a) Stocker dans le casier où il y a qu'un seul type de produits (identique au produit en cours de stockage). Si un seul casier, Stocker Si plusieurs casiers, Stocker dans le casier où il y a le plus de produit de ce type.
Si un seul casier, Stocker Sinon, Stocker dans le casier le plus proche de la station de déstockage.
Si un seul casier, Stocker Sinon, Stocker dans le casier le plus proche du point de repos de la machine de déstockage.
Sinon,
 - (b) Stocker dans le casier où il y a le plus de produits de même type (identique au produit en cours de stockage). Si un seul casier, Stocker Sinon, Stocker dans le casier où il y a le moins de produits en général.
Si un seul casier, Stocker Sinon, Stocker dans le casier le plus proche de la station de déstockage.
Si un seul casier, Stocker Sinon, Stocker dans le casier le plus proche du point de repos de la machine de déstockage. Sinon,
2. Stocker dans un casier vide. Si un seul casier, Stocker Si plusieurs casiers, Stocker dans le casier le plus proche de la station de déstockage.
Si un seul casier, Stocker Sinon, Stocker dans le casier le plus proche du point de repos de la machine de déstockage.
Sinon,
3. Stocker dans le casier où il y a le moins de produits. Si un seul casier, Stocker Sinon, Stocker dans le casier où il y a le moins de types de produits différents.
Si un seul casier, Stocker Sinon, Stocker dans le casier le plus proche de la station de déstockage.
Si un seul casier, Stocker Sinon, Stocker dans le casier le plus proche du point de repos de la machine de déstockage.

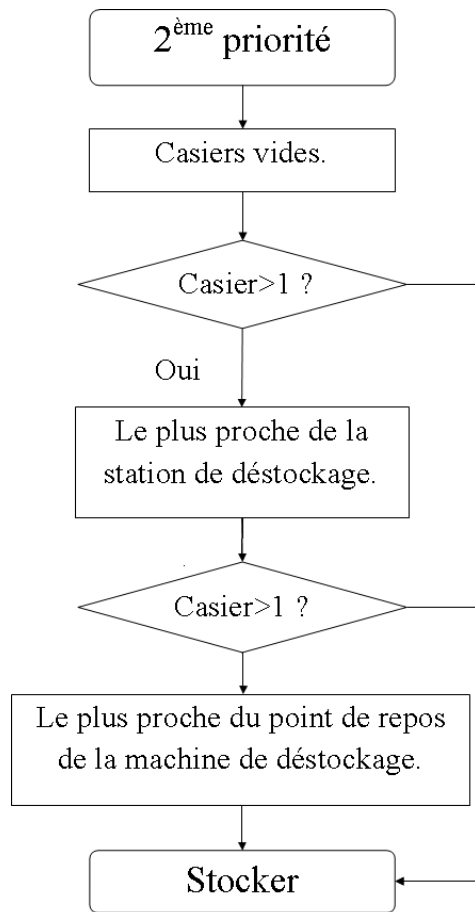
Les trois priorités sont représentées dans les organigrammes 2.2, 2.3 et 2.4.

II.3 Heuristique pour le stockage/déstockage dans un AS/RS à convoyeur gravitationnel

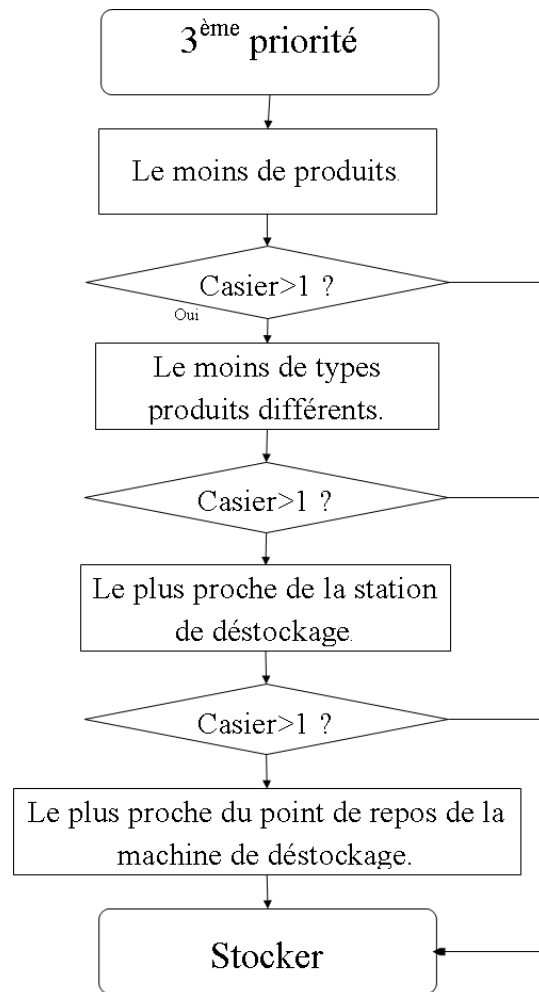


Organigramme 2.2

II.3 Heuristique pour le stockage/déstockage dans un AS/RS à convoyeur gravitationnel



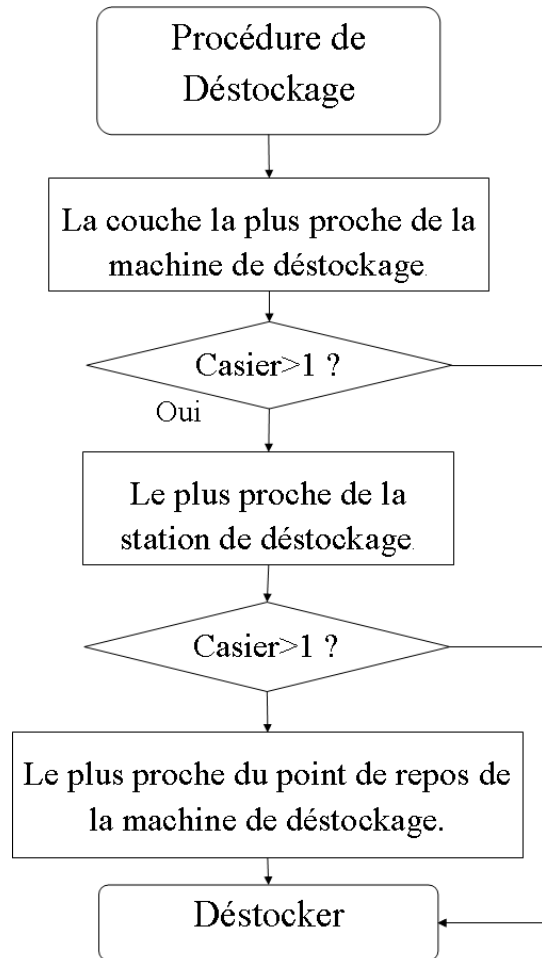
Organigramme 2.3



Organigramme 2.4

3.2 Procédure de déstockage

Chercher le produit désiré se trouvant dans la couche la plus proche de la machine de déstockage. Si un seul casier, Déstocker Sinon, Déstocker à partir du casier le plus proche de la station de déstockage. Si un seul casier, Déstocker Sinon, Déstocker à partir du casier le plus proche du point de repos de la machine de déstockage. La procédure de déstockage est représentée dans l'organigramme 2.5.



Organigramme 2.5

4 Conclusion

Dans ce chapitre, nous avons traité dans la première partie les différents aspects du système de stockage/déstockage à convoyeur gravitationnel, composantes, fonctionnement ainsi que les différents modèles analytiques. Dans la seconde partie nous avons présenté l'heuristique de stockage/déstockage proposé par Sari [ZS05] et amélioré par Gaouar *et al* [FEM04a].

Gaouar *et al* ont déduit que l'heuristique de stockage/déstockage a permis à l'AS/RS à convoyeur gravitationnel une grande amélioration en temps de cycle et en variété de produits, ainsi ses performances globales sont devenues supérieures à celles de l'AS/RS à convoyeur gravitationnel utilisant le stockage/déstockage aléatoire et donc l'heuristique a permis à l'AS/RS à convoyeur gravitationnel de devenir une alternative intéressante à l'AS/RS à charge unitaire.

Chapitre III

Les Méthaheuristiques

1 Introduction

Les Méthaheuristique sont des méthodes d'optimisation approchées qui fournissent des résultats satisfaisants en un temps de calcul aussi faible que possible à des problèmes combinatoires réputés difficiles pour lesquels on ne connaît pas de méthode exacte ou que cette dernière nécessite un temps de calcul élevé.

Nous présentons dans ce chapitre les principes fondamentaux des deux Méthaheuristiques qui constituent l'objet de ce mémoire : le Recuit Simulé et la Recherche Tabou. Nous donnons d'abord, un aperçu général sur les Méthaheuristiques. Ensuite, nous présentons les principes fondamentaux des deux méthodes (le recuit simulé SA, la recherche tabou TS) en citant leurs définitions, leurs origines et leurs algorithmes de bases.

2 Méthaheuristiques

2.1 Définition

Le mot méthaheuristique est dérivé de la composition de deux mots grecs :

- heuristique qui vient du verbe heuriskein et qui signifie 'trouver'
- méta qui est un suffixe signifiant 'au-delà', 'dans un niveau supérieur'.

Avant l'adoption de ce mot, on parlait plutôt d'heuristique moderne. Plusieurs définitions ont été proposées pour expliquer clairement ce qu'est une méthaheuristique, parmi lesquelles :

"A metaheuristic is formally defined as an iterative generation process which guides a subordinate heuristic by combining intelligently different concepts for exploring and exploiting the search space, learning strategies are used to structure information in order to find efficiently near-optimal solutions." [IO96]

2.2 Formulation du problème

Soit S un ensemble de solutions à un problème d'optimisation.

Soit f une fonction permettant d'évaluer chaque solution s dans S .

Il s'agit de déterminer une solution $s^* \in S$ qui minimise f . Le problème à résoudre est donc le suivant :

$$\begin{aligned} \min f(s) \\ \text{s.c. } s \in S \end{aligned}$$

Une **structure de voisinage** est une fonction N qui associe un sous-ensemble de S à toute solution $s \in S$. une solution $s' \in N(s)$ est dite voisine de s .

Une solution $s' \in S$ est un **minimum local** relativement à la structure de voisinage N si $f(s') \leq f(s)$ pour tout $s \in N(s)$.

Une solution $s^* \in S$ est un **minimum global** si $f(s^*) \leq f(s)$ pour tout $s \in S$.

2.3 propriétés fondamentales des métaheuristiques

les propriétés fondamentales des métaheuristiques sont les suivantes.

- Les métaheuristiques sont des stratégies qui permettent de guider la recherche d'une solution optimale
- Le but visé par les métaheuristiques est d'explorer l'espace de recherche efficacement afin de déterminer des solutions (presque) optimales.
- Les techniques qui constituent des algorithmes de type métaheuristique vont de la simple procédure de recherche locale à des processus d'apprentissage complexes.
- Les métaheuristiques sont en général non déterministes et ne donnent aucune garantie d'optimalité
- Les métaheuristiques peuvent contenir des mécanismes qui permettent d'éviter d'être bloqué dans des régions de l'espace de recherche.
- Les concepts de base des métaheuristiques peuvent être décrit de manière abstraite, sans faire appel à un problème spécifique.
- Les métaheuristiques peuvent faire appel à des heuristiques qui tiennent compte de la spécificité du problème traité, mais ces heuristiques sont contrôlées par une stratégie de niveau supérieur.

- Les métaheuristiques peuvent faire usage de l'expérience accumulée durant la recherche de l'optimum, pour mieux guider la suite du processus de recherche.

2.4 classifications des métaheuristiques [Her05]

Il existe plusieurs façons de classer les métaheuristiques. On peut faire la différence entre les métaheuristiques qui s'inspirent de phénomènes naturels et celles qui ne s'en inspirent pas. Par exemple, les algorithmes génétiques et les algorithmes des fourmis s'inspirent respectivement de la théorie de l'évolution et du comportement de fourmis à la recherche de nourriture. Par contre, la méthode Tabou n'a semble-t-il pas été inspirée par un phénomène naturel. Une telle classification ne semble cependant pas très utile et est parfois difficile à réaliser. En effet, il existe de nombreuses métaheuristiques récentes qu'il est difficile de classer dans l'une des deux catégories. Certains se demanderont par exemple si l'utilisation d'une mémoire dans la méthode Tabou n'est pas directement inspirée de la nature [Her05].

Une autre façon de classer les métaheuristiques est de distinguer celles qui travaillent avec une population de solutions de celles qui ne manipulent qu'une seule solution à la fois. Les méthodes qui tentent itérativement d'améliorer une solution sont appelées méthodes de recherche locale ou méthodes de trajectoire. La méthode Tabou [Glo89],[Glo90], le Recuit Simulé [NM53], la Recherche à Voisinages Variables [NM97] sont des exemples typiques de méthodes de trajectoire. Ces méthodes construisent une trajectoire dans l'espace des solutions en tentant de se diriger vers des solutions optimales. L'exemple le plus connu de méthode qui travaille avec une population de solutions est l'algorithme génétique [Hol92].

Les métaheuristiques peuvent également être classées selon leur manière d'utiliser la fonction objectif. Etant donné un problème d'optimisation consistant à minimiser une fonction f sur un espace S de solutions, certaines métaheuristiques dites statiques travaillent directement sur f alors que d'autres, dites dynamiques, font usage d'une fonction g obtenue à partir de f en ajoutant quelques composantes qui permettent de modifier la topologie de l'espace des solutions, ces composantes additionnelles pouvant varier durant le processus de recherche [Her05].

Certaines métaheuristiques font usage de l'historique de la recherche au cours de l'optimisation, alors que d'autres n'ont aucune mémoire du passé. Les algorithmes sans mémoire sont en fait des processus markoviens puisque l'action à réaliser est totalement déterminée par la situation courante. Les métaheuristiques qui font usage de l'historique de la recherche peuvent le faire de diverses manières. On différencie généralement les méthodes ayant une mémoire à court terme de celles qui ont une mémoire à long terme [Her05].

Finalement, mentionnons que certaines métaheuristiques utilisent les concepts additionnels que

sont la diversification et l'intensification. Par diversification, on sous-entend généralement une exploration assez large de l'espace de recherche, alors que le terme intensification vient plutôt mettre l'accent sur l'exploitation de l'information accumulée durant la recherche. Il est important de bien doser l'usage de ces deux ingrédients afin que l'exploration puisse rapidement identifier des régions de l'espace de recherche qui contiennent des solutions de bonne qualité, sans perdre trop de temps à exploiter des régions moins prometteuses[Her05].

3 Méthode du recuit simulé (simulated annealing)[JD03]

Le recuit simulé est une métaheuristique inspirée d'un processus utilisé en métallurgie. Ce processus alterne des cycles de refroidissement lent et réchauffage (recuit) qui tendent à minimiser l'énergie du matériau. Cette méthode a été conçue simultanément par S. Kirkpatrick, C. D. Gellat et M. P. Vechi d'une part et par V. Cerny en Slovaquie d'autre part en 1982.

3.1 Analogie avec la thermodynamique

S. Kirkpatrick et ses collègues étaient des spécialistes de physique statique (qui s'intéressaient précisément aux configurations de basse énergie de matériaux magnétique désordonnés, regroupés sous le terme de verre de spin). La détermination numérique de ces configurations posait de redoutables problèmes d'optimisation, car le « paysage d'énergie » d'un verre de spin présente une multitude de vallées de profondeurs inégales ; il est analogue au « paysage » de la figure III.1 .Il ont alors, proposé de traiter ces problèmes en s'inspirant de la technique expérimentale du recuit utilisée par les métallurgistes pour obtenir un état solide « bien ordonné », d'énergie minimale (en évitant les structures « métastable », caractéristiques des minimums locaux d'énergie(Cn)). Cette technique consiste à porter le matériau à haute température, puis à abaisser lentement celle-ci. A titre d'illustration, nous représentons sur la figure ?? l'effet de la technique du recuit, et celui de la technique opposée de la trempe, sur un système formé d'un ensemble de particules. La méthode du recuit simulé transpose le procédé du recuit a la résolution d'un problème d'optimisation : la fonction objectif du problème, analogue a l'énergie d'un matériau, est alors minimisée, moyennant l'introduction d'une température fictive, qui est, dans ce cas, un simple paramètre de contrôle de l'algorithme. Donc l'idée général du recuit simulé est d'exploiter l'algorithme de metropolis, qui permet de décrire le comportement d'un système en « équilibre thermodynamique » à une certaine température T : a partir d'un état initial, on fait une modification élémentaire, si elle améliore la fonction objectif elle est

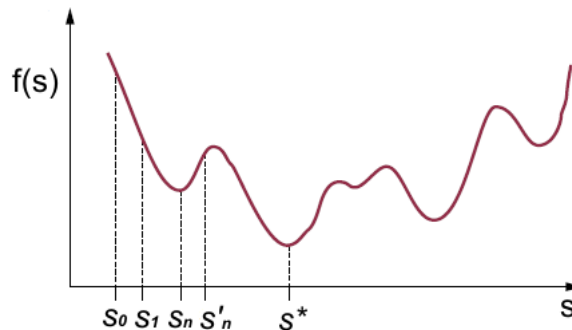


Figure III.1 – Fonction objectif d’un problème d’optimisation difficile en fonction de la configuration [JD03]

acceptée, par contre si elle dégrade la fonction objectif elle peut être acceptée avec la probabilité $\exp(-\nabla E/T)$ ou $\nabla E = f(s') - f(s)$ on itère ce procédé, en gardant la température constante, jusqu’à ce que l’équilibre thermodynamique soit atteint, concrètement au bout d’un nombre « suffisant » de modifications. On abaisse alors la température, avant d’effectuer une nouvelle série de transformation : la loi de décroissance par paliers de la température est souvent empirique, tout comme le critère d’arrêt du programme.

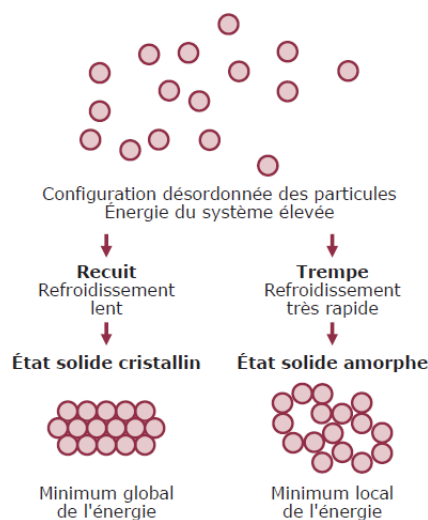


Figure III.2 – comparaison des technique du recuit et de la trempe [JD03]

3.2 Algorithme du recuit simulé

l’organigramme de l’algorithme du recuit simulé est représenté par la figure III.3.

Algorithm 1 recuit simulé

```

Début
Déterminer la solution initiale  $S_0$  et la température initiale  $T_0$ ;
Poser  $s = S_0$  et  $T_i = T_0$ ;
Calculer  $f(s_0)$ ;
Tant qu'aucun critère d'arrêt n'est satisfait faire
    Choisir  $s' \in N(s_i)$ ;
    Calculer  $\Delta f = f(s') - f(s)$ ;
    Si  $\Delta f < 0$  Alors  $s = s'$ ;
    sinon
        Début
        tirer  $p$  dans  $[0,1]$  suivant une distribution uniforme;
        Si  $p \leq \exp \frac{-\Delta f}{T}$  : Alors  $s = s'$ ;
        Sinon  $s'$  est rejetée;
        Fin;
    mettre à jours  $T$ ;
Fin Tant que
Fin.
```

3.3 Réglage des paramètres de la méthode du Recuit Simulé

Afin que le calcul converge le plus rapidement possible il est nécessaire de bien régler certains paramètres

- La configuration initiale est soit choisie aléatoirement ou avec un algorithme glouton pour démarrer avec une température initiale basse
- La température de départ détermine le temps de refroidissement pour atteindre la température d'arrêt et le nombre de paliers de température

La diminution de la température peut être effectuée par différentes fonctions. La méthode la plus utilisée consiste à définir $T_{k+1} = \alpha T_k$ où α est un nombre réel choisi entre 0 et 1 (typiquement $\alpha=0.95$), on peut aussi définir $T_{k+1} = \frac{L}{\log K+c}$ où L et c sont deux constante, mais de telles stratégies de décroissance de la température sont trop lentes et nécessitent donc des temps de calcul astronomiques avant d'atteindre la convergence vers l'optimum global. La décroissance de la température peut également être réalisées par paliers. Certains préconisent l'utilisation de stratégies non monotones. On peut ainsi rehausser la température lorsque la recherche semble bloquée dans une région de l'espace de recherche. On peut alors considérer une grande augmentation de la température comme un processus de diversification alors que la décroissance de la température correspond à un processus d'intensification.

- Condition d'arrêt : l'algorithme s'arrête si :

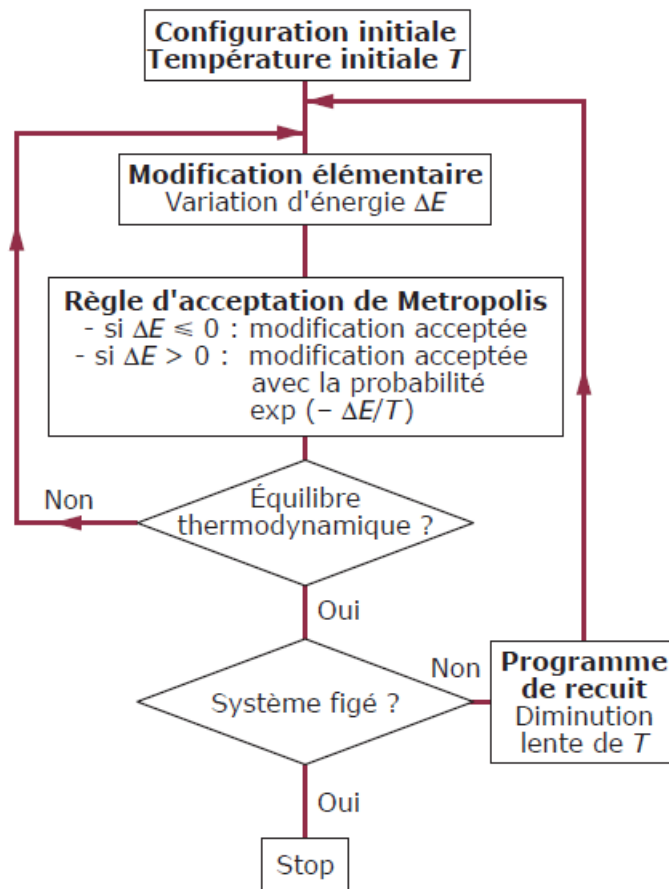


Figure III.3 – Organigramme de l'algorithme du recuit simulé

1. il a atteint un nombre maximal d'itération
2. la température a atteint un seuil assez bas
3. le système devient figé c'est-à-dire qu'il y a convergence vers une solution

4 La Méthode de recherche avec tabous (Tabu Search)[JD03]

La méthode de recherche avec tabous (ou simplement recherche tabou ou méthode tabou) a été formalisée en 1986 par F. Glover [Glo89]. Sa principale particularité tient dans la mise en oeuvre de mécanismes inspirés de la mémoire humaine. La méthode tabou prend, sur ce plan, le contre-pied du recuit simulé, totalement dépourvu de mémoire, et donc incapable de tirer les leçons du passé.

4.1 Principe de la méthode Tabou

Le principe de la Recherche Tabou est de choisir à chaque itération la meilleure solution $s' \in N(s)$, même si $f(s') > f(s)$. Lorsqu'on atteint un minimum local s par rapport au voisinage N , la Recherche Tabou va donc se déplacer vers une solution s' plus mauvaise que s . Le danger est alors de revenir à s immédiatement si $s \in N(s')$ puisque s est meilleure que s' . Pour éviter de tourner ainsi en rond, on crée une liste LT qui mémorise les dernières solutions visitées et interdit tout déplacement vers une solution de cette liste. Cette liste LT est appelée **liste taboue**, Les solutions ne demeurent dans LT que pour un nombre limité d'itérations. La liste LT est donc une mémoire à court terme. Si une solution s' est dans LT on dit que s' est une solution taboue. De même, tout mouvement qui nous mène de la solution courante à une solution de LT est appelé mouvement tabou.

4.2 Algorithme de la recherche Tabou

Algorithm 2 Recherche Tabou

Début

- 1 Déterminer la solution initiale $S_0, s^* = s, k = 0, LT = \emptyset$; (LT : Liste Taboue)
- 2 $K := K + 1$; générer un sous-ensemble de solutions en $N(s, k)$ pour que :
 - les mouvements tabous ne soient pas choisis
- 3 choisir la meilleure solution s' parmi l'ensemble de solutions voisines $N(s, k)$
- 4 actualisation de la meilleure solution connue $s := s'$;
- 5 Si $f(s') \leq f(s)$ alors nous avons trouvé une meilleure solution $s^* := s$;
- 6 mettre à jours la liste TL
- 7 Si une condition d'arrêt est atteinte, stop
Sinon aller à l'étape 2

Fin.

l'organigramme de la recherche avec tabou est représenté par la figure III.4.

4.3 Améliorations de la méthode de la recherche tabou

La recherche de la solution optimale peut être améliorée. Voici quelques options :

- Choix stratégique de la solution initiale s (exemple : algorithme glouton). ceci donnera une « bonne » valeur de $f(s^*)$.
- Intensification : La recherche est menée dans un voisinage $N(s)$ de S , l'ensemble des solution, une haute priorité est donnée aux solution $f(s')$ qui ressemble a la solution

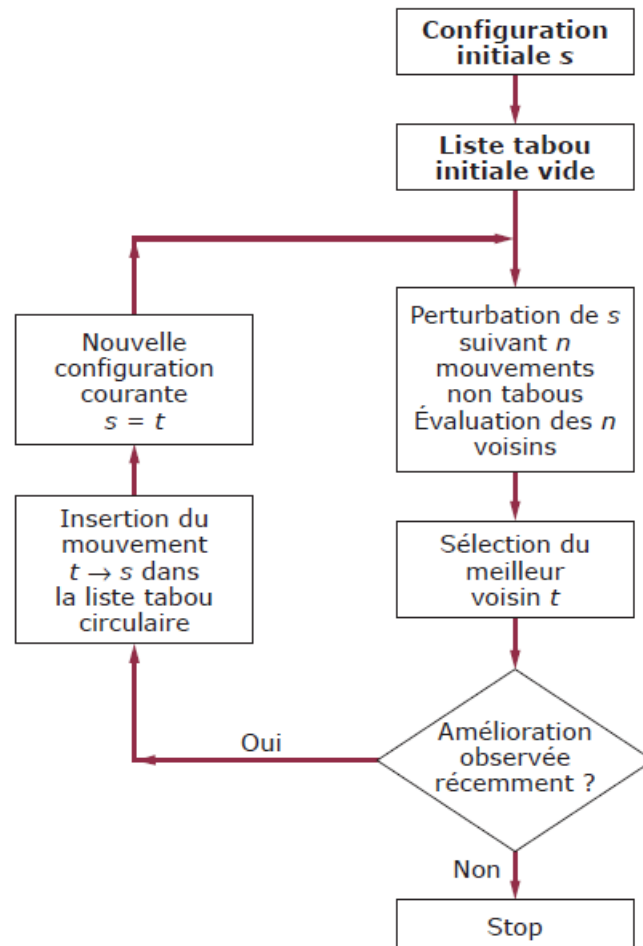


Figure III.4 – Organigramme de la Recherche avec tabou

actuelle $f(s)$ qui donne de meilleurs résultats ; le résultat est donc une intensification de la recherche dans un certain secteur, dans un voisinage choisi.

- Diversification : La recherche est éloignée du voisinage $N(s)$ actuel de l'ensemble des solutions, une priorité est donnée aux solutions $f(s')$ d'une autre région que celle actuellement sous exploration, le résultat : chercher ailleurs
- Sélection du meilleur voisin :
 - Best fit : le voisinage est exploré en entier
 - First fit : une partie du voisinage est explorée (voisinage très grand, évaluation difficile).
- Aspiration : Consiste à lever le statut Tabou d'un mouvement, s'il se révèle intéressant, en général, le mouvement est choisi quelque soit son état s'il conduit à une amélioration

de la meilleure solution

5 Conclusion

Les métaheuristiques font l'objet d'une recherche intense pour la résolution de divers problèmes NP-difficile. Elles ont révélé leur grande efficacité de fournir des solutions approchées de bonne qualité pour un grand nombre de problèmes. La présentation générale menée au cours de ce chapitre sur le recuit simulé et la recherche tabou portant sur leur définition et l'explication de leur principe de fonctionnement ont permis de faire ressortir l'idée de base de chacune de ces métaheuristiques pour leur adaptation et application au problème d'optimisation du temps de déstockage dans un AS/RS à convoyeur gravitationnel.

Chapitre IV

Simulateur d'AS/RS à convoyeur gravitationnel

1 Introduction

Dans le but de résoudre le problème d'optimisation du temps de déstockage dans un système AS/RS à convoyeur gravitationnel par les deux métaheuristique, recuit simulé et recherche tabou, nous avons développé un simulateur d'AS/RS à convoyeur gravitationnel, afin d'implémenter et d'évaluer les performance des deux métaheuristicques.

2 Introduction à la simulation

2.1 Définition

La simulation, selon ROBERT E. SHANNON [SHA98], est « le processus de concevoir un modèle d'un vrai système et d'entreprendre des expériences avec ce modèle dans le but de comprendre le comportement du système ou d'évaluer diverses stratégies (dans les limites imposées par un critère ou par un ensemble de critères) pour le fonctionnement du système ». Le but de la simulation est de comprendre le comportement dynamique du système, de comparer des configurations, d'évaluer différentes stratégies de pilotage, d'évaluer et d'optimiser des performances.

2.2 Méthodologie d'une simulation

On distingue classiquement quatre phases distinctes : la modélisation (représenter le comportement du système), la programmation, l'expérimentation et l'interprétation des résultats (accompagnée d'actions).

- La modélisation : est le processus de création d'un modèle conceptuel qui décrit le fonctionnement du système réel avec le degré de détail nécessaire à la résolution du problème

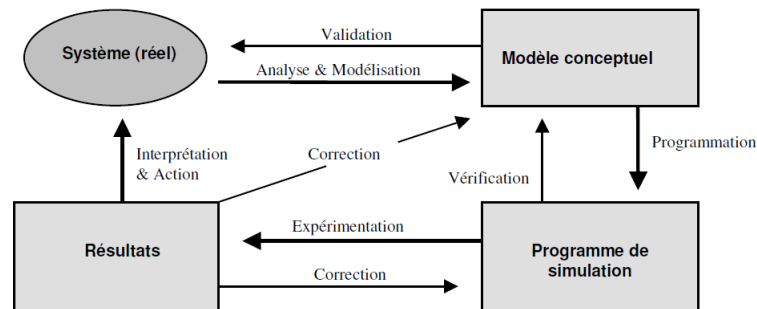


Figure IV.1 – Méthodologie d'une simulation

posé.

- La programmation : est le passage du modèle conceptuel au programme de simulation en utilisant les langages de programmation ou les logiciels de simulation
- L'expérimentation : il s'agit de construire des théories, ou hypothèse, qui prennent en compte le comportement observé.
- L'interprétation des résultats : nous permet de mieux comprendre le comportement du système étudié.

2.3 Avantages et inconvénients de la simulation

Les avantages de la simulation peuvent être résumés comme suit :

- La simulation est non destructrice. Les erreurs ne sont pas coûteuses.
- Le système considéré n'a même pas besoin d'exister.
- On peut répéter à volonté des expériences identiques ou similaires dans les mêmes conditions.
- On peut souvent simuler un système beaucoup plus rapidement que son évolution dans la réalité (exemple : simulation de l'évolution d'un biosystème).
- On peut simuler des modèles très complexes, plus réalistes que ceux que l'on peut résoudre par des formules analytiques ou par les méthodes d'optimisation classique.
- L'animation graphique peut permettre de voir évoluer le modèle.

Elle n'est cependant pas sans inconvénients :

- Coût : la modélisation et la programmation peuvent demander beaucoup d'effort de temps et d'argent.
- Les temps d'exécution (CPU) peuvent devenir excessifs.
- Ne fournit habituellement que des estimations ; l'optimisation est beaucoup plus difficile

par simulation que via les outils habituels de programmation mathématique.

- L'analyse statistique des résultats n'est pas toujours simple.

2.4 Simulation de systèmes à événements discrets

Les systèmes à événements discrets, qui modélisent des phénomènes se déroulant en temps réel, mais dont les évolutions se produisent en des instants ponctuels, constituent une classe importante de modèles stochastiques, et interviennent dans de nombreux domaines d'application, la simulation étant souvent un moyen privilégié pour leur étude. Une méthode générale pour concevoir et réaliser ce type de simulation est celle du **déroulement en temps (pseudo-)réel** concrètement, cela signifie que la simulation se déroule en faisant sauter un compteur de temps, d'un événement affectant l'évolution du système à celui qui le suit immédiatement, tout en mettant à jour à chaque saut les différentes variables caractérisant l'état du système. Ceci nécessite de tenir à jour une liste d'événements contenant, pour chaque type d'événement possible, la liste des dates des prochains événements programmés de chacun des types.

2.5 Logiciels de simulation

Le choix d'un logiciel de simulation est une étape importante pour mener à bien l'étude d'un modèle. Si le logiciel choisi n'est pas assez flexible ou trop difficile à utiliser, le projet de simulation peut produire des résultats erronés, voire tout simplement ne pas être conduit à terme. Une analyse du choix du logiciel de simulation est conduite dans Law [Law07], Deux tendances s'opposent en simulation, à savoir l'utilisation d'un packaging de simulation et celle d'un langage de programmation classique. Chaque approche a ses avantages et ses inconvénients, comme résumé ci-dessous :

1. Packaging de simulation :

- fournissent la plupart des caractéristiques nécessaires pour construire un modèle de simulation ;
- environnement naturel pour la simulation ;
- modèles de simulation généralement plus faciles à écrire et à modifier ;
- généralement beaucoup de vérification d'erreurs intégrées (facilite la tâche de vérification) ;
- on distingue les packaging de simulation à but générique, et les packagings de simulations orientés pour des applications particulières.

2. Langage de programmation :

- beaucoup de modélisateurs connaissent déjà un langage de programmation ;
- un modèle de simulation en C, C++, Java, . . . peut être plus rapide à exécuter ;
- plus de flexibilité de programmation ;
- on peut bénéficier de l'approche orienté-objet ;
- le logiciel de développement coûte moins cher, bien que le coût total du projet puisse être plus important étant donné que le temps de développement en particulier sera souvent plus long.

Si nous nous intéressons plus particulièrement aux paquetages de simulation, nous pouvons en outre opérer une distinction entre les logiciels de simulation à but générique et ceux conçus pour des applications particulières. Historiquement, les langages de simulation offraient un meilleur niveau de flexibilité, mais étaient souvent difficile à utiliser. A contrario, les simulateurs orienté application bénéficiaient d'avantage d'interfaces graphiques, étaient plus facile d'usage, mais pouvaient manquer de souplesse pour certains problèmes. Ces distinctions tend toutefois à disparaître comme les langages de simulations sont à présent souvent intégrés dans des environnements graphiques, et les logiciels de simulation gagnent quant à eux en flexibilité. La distinction majeure reste donc sur la classe des simulations que peut traiter un logiciel donné, variant entre une orientation vers des applications spécifiques, ou au contraire une approche plus générique, même si le logiciel peut disposer en outre de caractéristiques spécifiques pour certaines applications.

3 Modélisation du Système AS/RS à convoyeur gravitationnel

L'élaboration d'un système informatique passe toujours par une phase de modélisation préalable afin de pouvoir simuler la réalité. Pour la modélisation du système AS/RS à convoyeur gravitationnel, nous utilisons la méthode de modélisation orientée objet.

La modélisation orientée objet est une méthode de modélisation caractérisée par la représentation naturelle qu'offre le concept d'objet pour les entités physiques ou conceptuelles du monde réel. La modélisation orientée objet consiste à représenter une entité sous la forme d'un objet, c'est-à-dire sous la forme d'une entité informatique qui rassemble, dans une structure de données abstraite, les données et les procédures qui sont nécessaires à sa représentation et sa gestion. Un objet représente une abstraction sur les données et les traitements exclusifs associés à ces données.

IV.3 Modélisation du Système AS/RS à convoyeur gravitationnel

dans notre projet nous avons utilisé le langage UML (Unified Modeling Language) comme moyen de modélisation. UML est un langage de modélisation orienté objet normalisé par l'OMG (Object Management Group) et adopté par les méthodes d'analyse et conception orientées objets existantes. L'UML consiste en un ensemble de notations graphiques et sémantique qui sont bien définis. Il permet de dépeindre un domaine particulier comme un système orienté objet (dans notre cas, système AS/RS à convoyeur gravitationnel). La documentation qui traite le sujet de présentation d'UML est largement disponible et très riche, de ce fait, dans ce qui suit nous n'allons pas présenter, mais plutôt utiliser le langage UML pour la présentation conceptuelle des composants du cadre de simulation proposé. nous avons décomposé le système en plusieurs classes, qui sont illustrés dans la figure IV.2. Dans la partie qui suit, nous détaillerons les principales classes.

IV.3 Modélisation du Système AS/RS à convoyeur gravitationnel

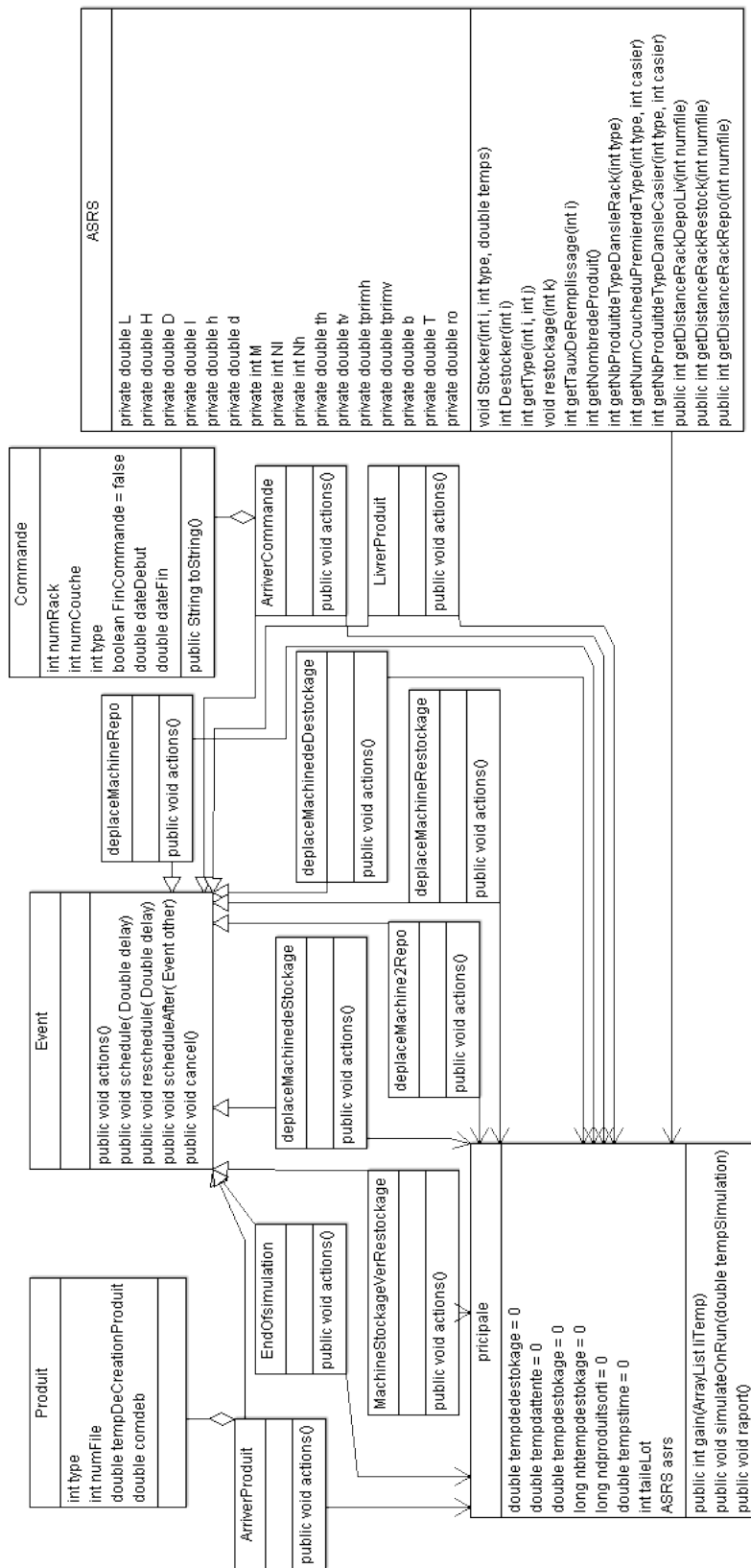


Figure IV.2 – Diagramme des classes

3.1 description des principales classes

1. **Classe ASRS()** : cette classe représente le rack profond

(a) attribut :

- L,H,D :longueur, hauteur et profondeur du rack AS/RS à CG ;
- longueur, hauteur et profondeur d'un emplacement de stockage ;
- Nl : nombre de casier pour chaque ligne ;
- Nh : nombre de casier pour chaque colonne ;
- M : nombre d'emplacement dans un casier, nombre de couches dans un rack ;
- th : temps de transport horizontal de la station de dépôt/livraison à la colonne la plus éloignée ;
- tv : temps de transport vertical de la station de dépôt/livraison à la ligne la plus éloignée ;
- tprimh : temps de transport horizontal entre deux casiers mitoyen ;
- tprimv : temps de transport vertical entre deux casiers mitoyen ;
- T : facteur de normalisation ;
- ro :taux de charge ;
- b :facteur de forme ;

(b) méthodes :

- stocker(int i,int type,double temps) : stocker le produit de type type dans le casier numéro i
- destocker(int i) : déstocker du casier numéro i
- getType(int i, int j) : renvoi le type du produit qui se trouve dans le casier i et dans la couche j
- getTauxDeRemplissage(int i) : renvoi le taux de remplissage dans le casier i

2. **Classe Event()** : c'est une abstraction de base qui représente un événement général dans une simulation à événements discrets.

(a) méthodes :

- actions() : C'est la méthode qui est exécutée lorsque cet événement se produit. Chaque sous-classe de l'événement qui doit être instancié doit fournir une implémentation de cette méthode.
- schedule(double delay) : cet événement se produise dans delay unités de temps, c'est à dire, à $\text{sim.time} + \text{delay}$.
- scheduleAfter(Event other) :cet événement se produise juste après l'événement other

- `cancel()` : Annule cet événement avant qu'il se produise. Retourne vrai si l'annulation parvient.
 - `reschedule(double delay)` : annule cet événement et le reprogramme dans `delay` unités de temps.
3. **Classe `ArriverCommande()`** : cette classe hérite de la classe `EVENT` et elle représente l'arrivée périodique des commandes.
 4. **Classe `ArriverProduit()`** : cette classe hérite de la classe `EVENT` et elle représente l'arrivée périodique des produits.
 5. **Classe `MachineStockageVerRestockage()`** : cette classe hérite de la classe `EVENT` et elle représente le déplacement de la machine de stockage vers le point de restockage (convoyeur de restockage)
 6. **Classe `deplaceMachinedeDestockage`** : cette classe hérite de la classe `EVENT` et elle représente le déplacement de la machine de destockage
 7. **Classe `deplaceMachinedeStockage`** : cette classe hérite de la classe `EVENT` et elle représente le déplacement de la machine de stockage
 8. **Classe `deplaceMachineRepo`** : cette classe hérite de la classe `EVENT` et elle représente le déplacement de la machine de stockage vers la station de repos.
 9. **Classe `deplaceMachine2Repo`** : cette classe hérite de la classe `EVENT` et elle représente le déplacement de la machine de destockage vers la station de repos.
 10. **Classe `LivrerProduit`** : cette classe hérite de la classe `EVENT` et elle représente le déplacement de la machine de destockage vers la station de livraison.
 11. **Classe `EndOfsimulation`** : cette classe hérite de la classe `EVENT` et elle représente la fin de la simulation.

4 Implémentation

Pour le codage de notre simulateur, nous avons choisi le langage de programmation java et l'environnement de développement eclipse.

4.1 JAVA

Apparu fin 1995 début 1996 et développé par Sun Microsystems. Le langage Java a la particularité principale d'être portable sur plusieurs systèmes d'exploitation tel que Unix, Microsoft

Windows, Mac OS ou Linux... C'est la plateforme qui garantit la portabilité des applications développées en Java. Le langage reprend, en grande partie, la syntaxe du langage C++, très utilisée par les informaticiens. Néanmoins, Java a été épuré des concepts les plus subtils du C++ et à la fois les plus déroutants, tel que l'héritage multiple remplacé par l'implémentation des interfaces. Les concepteurs ont privilégié l'approche orientée objet de sorte qu'en Java, tout est objet à l'exception des types primitifs (nombres entiers, nombres à virgule flottante, etc.).

4.2 Environnement de développement ECLIPSE

Eclipse est un environnement de développement intégré (Integrated Development Environment) dont le but est de fournir une plate-forme modulaire pour permettre de réaliser des développements informatiques.

5 Validation du simulateur

Dans cette section nous présentons les résultats de la simulation de l'AS/RS à convoyeur gravitationnel par le simulateur que nous avons développé.

dans un premier lieu nous avons varié le nombre de casiers des lignes (Nl), le nombre de casiers des colonnes (Nh) et le nombre de couches du rack ; ensuite nous avons varié le taux de charge ρ par incrément de 10% de 20% à 70% et 5% de 70% à 95%, pour le système qui se caractérise par :

Nombre de casiers et de couches : $Nl = 20$, $Nh = 20$, $M = 10$.

Temps de déplacement : $t_h = 0.2$, $t_v = 0.2$

Facteurs de forme et normalisation : $b=1$, $T=0.2$

Le système a été simulé sur $10^5 s$. pour chaque configuration, la simulation s'est faite plusieurs fois et une valeur moyenne a été calculée. Les résultats des simulations ont été comparées à celle du modèle analytique [II.20](#).

Dans le tableau [V.1](#), sont représentés les résultats de la première simulation, ceux du modèle analytique ainsi que l'erreur relative entre les deux. Les résultats de la deuxième simulation sont représentés par le tableau [V.2](#) et la figure [IV.3](#). Nous remarquons à travers le tableau [V.1](#), [V.2](#) et la [IV.3](#) que les résultats de simulation sont très proches de ceux du modèle analytique et que l'erreur entre les deux est négligeable. Ceci valide notre simulateur.

IV.5 Validation du simulateur

Nl	Nh	M	E(RC) analytique	E(RC) simulation	Erreur relatives(%)
10	10	10	0.48	0.5	2.08
20	20	10	1.013	1.053	1.97
20	20	20	1.963	2.006	1.98
40	40	10	2.08	2.14	1.44
40	40	20	4.03	4.08	0.99
40	40	40	8.006	7.932	0.93
60	60	30	9.048	9.17	1.01
60	60	60	17.896	18.204	0.6

Tableau IV.1 – Temps moyen de déstockage de différent configuration d’AS/RS à convoyeur gravitationnel

Taux de charge (%)	20	30	40	50	60	70	75	80	85	90
Résultats analytique	0.316	0.443	0.569	0.696	0.823	0.95	1.013	1.076	1.14	1.203
Résultats de simulation	0.42	0.538	0.649	0.764	0.878	0.993	1.053	1.107	1.168	1.232
Erreurs relatives (%)	4.43	3.38	3.51	2.58	1.82	0.73	1.08	2.88	1.57	2.4

Tableau IV.2 – Temps moyen de déstockage d’un AS/RS à convoyeur gravitationnel

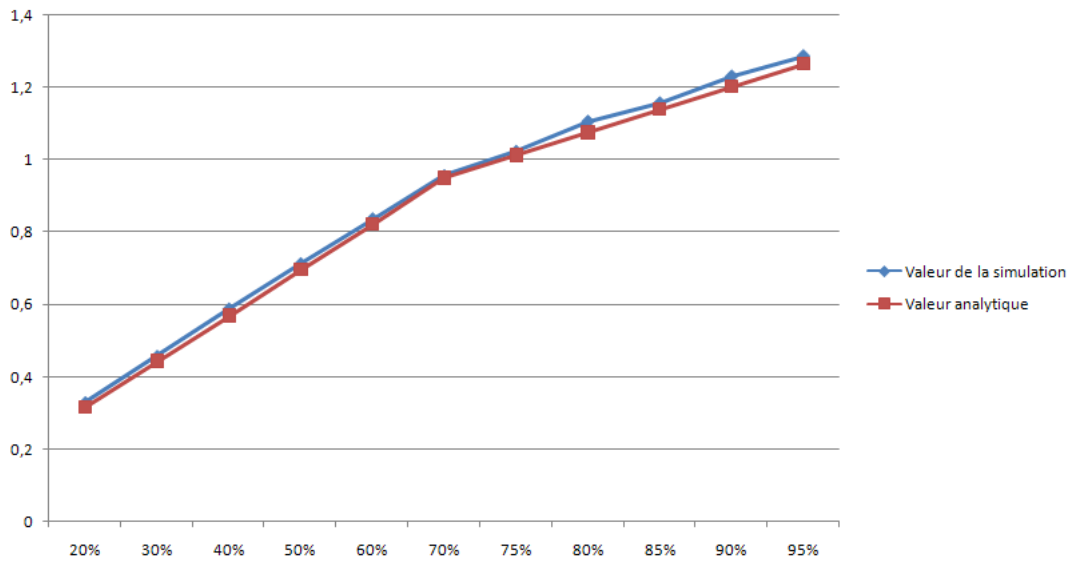


Figure IV.3 – Temps moyen de déstockage d’un AS/RS à convoyeur gravitationnel en fonction du taux de charge

6 Conclusion

Dans ce chapitre, nous avons discuté de la simulation, de ses avantages et inconvénients, ensuite nous avons présenté les principales classes réalisées et implémentées. Ce logiciel a été validé par la simulation et la comparaison des résultats avec le modèle analytique [II.20](#). dans le chapitre qui suit nous simulerons le système AS/RS à convoyeur gravitationnel, les méta-heuristique seront utilisés pour le destockage, pour le stockage nous utiliserons l'aléatoire et l'heuristique développé par Sari [[ZS05](#)].

Chapitre V

Adaptation du recuit simulé et la recherche tabou pour l'AS/RS à convoyeur gravitationnel

1 Introduction

L'objectif de ce chapitre est de présenter les algorithmes du recuit simulé et de la recherche tabou adaptés au problème d'optimisation du temps de déstockage d'AS/RS à convoyeur gravitationnel et de donner une synthèse des résultats obtenus. Afin d'évaluer les performances des Métaheuristique implémentés, les résultats obtenus sont comparés aux résultats des méthodes heuristiques classiques. Pour le réglage des paramètres des deux métaheuristiques, une étude de sensibilité des paramètres a été faite et les meilleurs paramètres ont été sélectionnés pour l'étude comparative.

2 Les algorithmes des métaheuristiques

Dans un AS/RS à convoyeur gravitationnel le déstockage d'un produit se trouve en position $m(m < M)$ s'effectue en déstockons tous les produit qui le précèdent et les renvoyer vers le convoyeur de restockage afin qu'ils soient restocké, donc Pour le destockage de deux produits P1 et P2, on déstocke tous les produits qui précèdent P1, ensuite, tous ceux qui précèdent P2, même si P2 se trouve parmi les produits qui précèdent P1 on va pas le livré, on ne prend en considération que le produit P1, pour y remédier a ce problème on a décidé de prendre un lot de produit et de trouver la meilleur séquence qui minimise le temps de destockage en utilisons les métaheuristiques.

l'algorithme du recuit simulé (Simulated Annealing SA) est comme suit :

Algorithm 3 recuit simulé

Début
 Déterminer la solution initiale s_0 , la température initiale T_i , la température finale T_f ;
 Poser $s = s_0$ et $T = T_i$;
 Calculer $f(s_0)$;
 Tant que $T \geq T_f$ faire
 Choisir $s' \in S$;
 Calculer $\Delta f = f(s') - f(s)$;
 Si $\Delta f < 0$ Alors $s = s'$;
 sinon
 Début
 tirer p dans $[0,1]$ suivant une distribution uniforme ;
 Si $p \leq \exp \frac{-\Delta f}{T}$: Alors $s = s'$;
 Sinon s' est rejetée ;
 Fin ;
 $T = \alpha T$;
 Fin Tant que
 Fin.

l'algorithme de la Recherche tabous (Taboo Search TS) est comme suit :

Algorithm 4 Recherche Tabou

Début
 Déterminer la solution initiale $s_0, s^* = s, i = 0$, le nombre d'itération $K, LT = \emptyset$;
 Tant que $i < K$ faire
 $i := i + 1$; générer un sous-ensemble de solutions en $N(s, i)$ pour que :
 - les mouvements tabous ne soient pas choisis
 choisir la meilleure solution s' parmi l'ensemble de solutions voisines $N(s, i)$
 actualisation de la meilleure solution connue $s := s'$;
 Si $f(s') \leq f(s)$ alors nous avons trouvé une meilleure solution $s^* := s$;
 mettre ajours la liste LT
 Fin Tant que
 Fin.

3 Etude de sensibilité

Pour cette étude, nous avons choisi un système AS/RS à convoyeur gravitationnel qui à les caractéristiques suivante :

V.3 Etude de sensibilité

Nombre de casiers et de couches : $Nl = 20$, $Nh = 20$, $M = 10$.

Temps de déplacement : $t'_h = 0.01$, $t'_v = 0.01$

Facteurs de forme et normalisation : $b=1$, $T=0.2$

t_i	t_f	α	S	E(RC)	$t_c(ms)$
36	0.1	0.5	7	0.2969	0.3658
44	0.1	0.5	7	0.2835	0.3874
52	0.05	0.5	11	0.2788	0.4403
44	0.1	0.6	7	0.2782	0.5513
52	0.05	0.5	7	0.2780	0.3341
36	0.1	0.5	11	0.2773	0.6086
52	0.1	0.6	7	0.2774	0.3428
44	0.05	0.5	10	0.2767	0.6041
40	0.1	0.5	10	0.2758	0.4986
52	0.1	0.5	10	0.2751	0.3822
36	0.1	0.8	10	0.2217	1.0497
28	0.001	0.6	10	0.2215	0.6049
28	0.1	0.8	10	0.2214	1.2093
44	0.001	0.6	10	0.2209	0.9765
44	0.005	0.7	10	0.2207	1.0323
44	0.05	0.7	11	0.2206	0.8971
28	0.01	0.7	9	0.2205	1.0516
40	0.01	0.7	10	0.2205	1.0860
28	0.001	0.9	9	0.1932	2.8835
40	0.001	0.9	7	0.1930	3.1258

t_i	t_f	α	S	E(RC)	$t_c(ms)$
28	0.005	0.9	13	0.1917	5.1319
28	0.005	0.9	11	0.1907	4.4331
28	0.05	0.9	10	0.1906	2.7439
28	0.01	0.9	9	0.1903	3.1773
28	0.01	0.9	10	0.1902	3.5843
44	0.005	0.9	11	0.1897	4.1399
52	0.01	0.9	10	0.1895	2.6037
52	0.001	0.9	11	0.1893	3.6943
36	0.001	0.9	13	0.1864	4.5590
28	0.001	0.9	10	0.1861	3.2445
44	0.001	0.9	13	0.1857	5.1016
40	0.005	0.9	11	0.1856	3.8967
36	0.001	0.9	9	0.1855	3.6923
28	0.001	0.9	13	0.1853	4.1399
36	0.01	0.9	13	0.1850	3.2537
28	0.01	0.9	13	0.1849	4.6045
40	0.001	0.9	13	0.1843	3.2662
52	0.001	0.9	13	0.1831	4.2756
40	0.001	0.9	11	0.1812	3.2186
52	0.001	0.9999	26	0.1810	6951.6

t_i : température initiale

t_f : température finale

α : coefficient de refroidissement

S : Equilibre thermodynamique

$E(RC)$: temps de déstockage moyen

$T_c(ms)$: temps machine d'un seul destockage

Tableau V.1 – résultat de l'étude de sensibilité du recuit simulé

Nous avons voulu vérifier l'influence des paramètres des deux métaheuristiques (recuit simulé et recherche tabou) sur la qualité des résultats. Pour cela nous avons fait varier les différents paramètres : Température initiale, température finale, équilibre thermodynamique et le coefficient de refroidissement α pour le recuit simulé. Le nombre d'itérations, nombre de voisins et la taille de la liste tabou pour la recherche tabou. La méthode du recuit simulé a donné un temps de destockage minimum lorsque le coefficient de refroidissement de la température

V.3 Etude de sensibilité

N_i	N_n	LT	$E(RC)$	$t_c(ms)$
1300	5	1	0.3137	8,9044
1000	20	1	0.3037	10,298
100	5	1	0.3025	0,8837
100	20	5	0.2979	2,9518
1000	1	1	0.2947	4,7566
1300	15	1	0.2924	13,426
1000	5	5	0.2367	19,271
1300	1	5	0.2365	16,818
1000	10	5	0.2361	40,267
1300	5	5	0.2360	26,397
100	20	10	0.2359	5,8049
100	1	15	0.2182	3,7220
100	20	15	0.2172	9,1860
100	15	15	0.2165	11,204
100	5	15	0.2141	9,0188
100	10	15	0.2092	11,810
100	5	20	0.2060	8,0674
100	10	20	0.2018	12,497
100	1	20	0.2016	8,6268

N_i	N_n	LT	$E(RC)$	$t_c(ms)$
100	20	20	0.1993	17,358
100	15	20	0.1983	11,387
1000	15	15	0.1900	125,91
1000	10	15	0.1892	78,947
1000	5	15	0.1881	70,850
1000	20	15	0.1850	136,03
1000	20	20	0.1848	204,45
1000	15	20	0.1839	163,15
1000	1	20	0.1839	70,445
1300	5	15	0.1837	61,943
1000	1	15	0.1836	57,085
1300	5	20	0.1836	81,376
1300	10	20	0.1825	134,41
1300	20	20	0.1824	251,41
1300	1	20	0.1818	93,927
1000	5	20	0.1816	109,31
1000	10	20	0.1815	118,21
1300	15	20	0.1800	214,17
1300	15	100	0.1785	6545.4

N_i : nombre d'itération

N_n : nombre de voisin

LT : taille de la liste tabou

$E(RC)$: temps de déstockage moyen

$T_c(ms)$: temps machine d'un seul destockage

Tableau V.2 – résultat de l'étude de sensibilité de la recherche tabou

$\alpha = 0.9$ comme illustré dans le tableau V.1, les autres paramètres ont une très petite influence ou pas d'influence du tout sur le temps de destockage. Pour vérifier l'influence de α nous avons fait une simulation avec $\alpha = 0.9999$, le temps de destockage n'a pas beaucoup changé mais le temps d'exécution machine est devenu trop grand. (voir dernière ligne du tableau V.1)

La recherche tabou a donné un temps de destockage minimal quand la taille de la file tabou est grande ($LT=20$) comme illustré dans le tableau V.2, les autres paramètres ont une petite influence sur le temps de destockage minimal. Pour vérifier l'impact de la taille de la file tabou sur le temps de destockage on fait une simulation avec une taille = 100, le temps de destockage n'a pas beaucoup changé mais le temps d'exécution machine a explosé. (voir dernière ligne du tableau V.2)

V.4 Analyse et comparaison des résultats de simulation

Taille du lot = 2			
N type produits	Recuit Simulé	recherche tabou	plus proche voisin
25	0.1866	0.2158	0.1878
50	0.2412	0.2475	0.2546
100	0.3242	0.3063	0.3252
400	0.5865	0.5712	0.5811
1000	0.7692	0.7553	0.7685
2000	0.9032	0.8962	0.9107
3000	0.9834	0.9649	0.9842
4000	1.0404	1.0250	1.0274
5000	1.0607	1.0516	1.0522

Tableau V.3 – temps de destockage des différentes méthodes par rapport au nombre de type produit pour un stockage aléatoire et un lot de taille 2

4 Analyse et comparaison des résultats de simulation

dans cette section nous allons analyser et comparer le temps de destockage de l'AS/RS à convoyeur gravitationnel obtenu par les deux métaheuristique recuit simulé et recherche tabou, avec les méthodes, heuristique de stockage/destockage, plus proche voisin et le modèle analytique de sari *et al.*2005. pour cela nous avons effectués plusieurs simulation en variant la taille des lots (tailles choisi 2,5 et 200) et le nombre des différents types de produits dans le système pour le déstockage et pour le stockage nous avons utilisé le stockage aléatoire et le stockage avec l'heuristique de sari 2003.

Les résultats de simulation sont représenté dans les tableaux ci-dessous. la première colonne représente le nombre de différents produits, les autres colonnes représentent respectivement le temps de destockage moyen obtenu par les méthode, recuit simulé, recherche tabou, plus proche voisin et le modèle analytique sari *et al.*2005.

Dans les figures [V.1](#),[V.2](#),[V.3](#),[V.4](#),[V.5](#) et [V.6](#), on remarque que le temps de destockage croit proportionnellement avec la diversification des produits quelles que soit la méthode de stockage ou de destockage utilisée, et il converge vers le résultat du destockage aléatoire quand le nombre des produits différents est assez important ; ceci se justifie par le fait que quand le nombre des produits différents est grand, tous les casiers du rack ont la même probabilité d'être utilisés pour le déstockage. Généralement, dans ce type de système le nombre total de produits différents dépasse rarement 20% de la capacité totale de stockage. Dans notre cas $N = 4000$ ce qui implique que le nombre des différent produit ne doit pas dépasser les 800. L'utilisation de l'heuristique de stockage donne de meilleur resultat quand le nombre de produits différents est petit, et elle améliore le temps de destockage par rapport au stockage aléatoire comme le

V.4 Analyse et comparaison des résultats de simulation

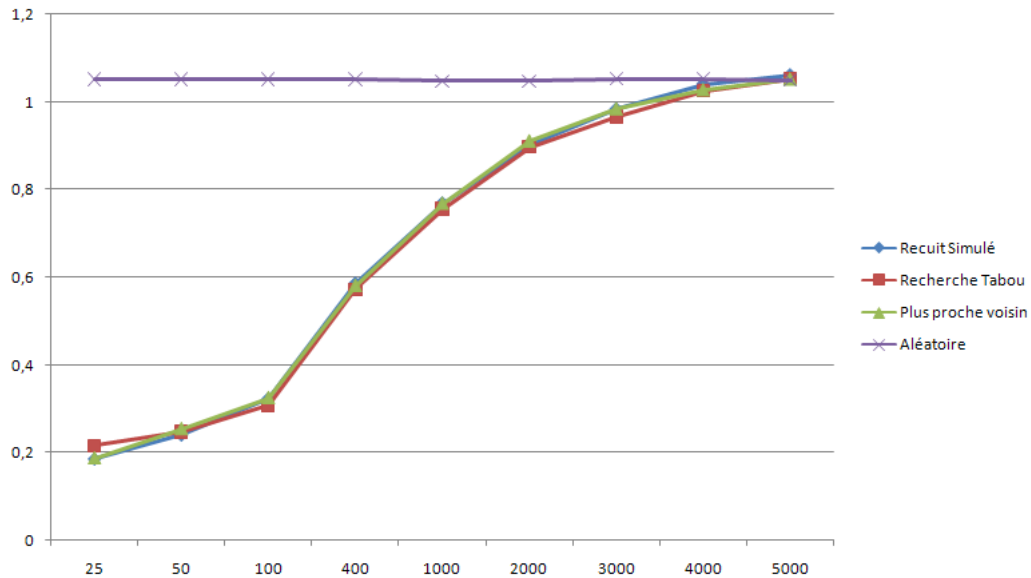


Figure V.1 – temps de destockage des différentes méthodes par rapport au nombre de type produit pour un stockage aléatoire et un lot de taille 2

montre les figures (V.1,V.2),(V.3,V.4) et (V.5,V.6). Les figures V.1 et V.2 montre que les trois méthode (recuit simulé, la recherche tabou et le plus proche voisin) sont presque identique quand la taille du lot est petite. les figures V.3 et V.4 montre que la recherche tabou donne de meilleurs résultats que le recuit simulé et le plus proche voisin quant la taille du lot et petite et le nombre de produit différent et assez petit, lorsque ce dernier augmente c la méthode du plus proche voisin qui donne des meilleurs résultats. les figures V.5 et V.6 montre que la méthode du plu proche voisin donne de meilleurs résultat lorsque la taille du lot est grande

V.4 Analyse et comparaison des résultats de simulation

Taille du lot = 2			
N°type produits	Recuit Simulé	recherche tabou	plus proche voisin
25	0.1153	0.1187	0.1289
50	0.1550	0.1606	0.1717
100	0.2124	0.2148	0.1993
400	0.4991	0.4945	0.5642
1000	0.7876	0.8031	0.7995
2000	0.9162	0.9094	0.9045
3000	1.0879	1.0359	1.0054
4000	1.0698	1.0192	1.0373
5000	1.0793	1.0863	1.0391

Tableau V.4 – temps de destockage des différentes méthodes par rapport au nombre de type produit pour un stockage avec heuristique et un lot de taille 2

Taille du lot = 5				
N produits	Recuit Simulé	Recherche Tabou	plus proche voisin	aléatoire
25	0.1952	0.1798	0.2830	1.0516
50	0.2259	0.2218	0.3233	1.0515
100	0.3363	0.3001	0.4236	1.0519
400	0.6105	0.5637	0.5848	1.0528
1000	0.7789	0.7466	0.7341	1.0483
2000	0.9097	0.8952	0.8511	1.0483
3000	0.9788	0.9763	0.9204	1.0529
4000	1.0222	1.0199	0.9656	1.0529
5000	1.0535	1.0536	0.9977	1.0505

Tableau V.5 – temps de destockage des différentes méthodes par rapport au nombre de type produit pour un stockage aléatoire et un lot de taille 5

Taille du lot = 5				
N°type produits	Recuit Simulé	recherche tabou	plus proche voisin	aléatoire
25	0.1487	0.1344	0.1462	1.0335
50	0.1871	0.1536	0.1666	1.0360
100	0.2365	0.2025	0.2072	1.0348
400	0.6116	0.4813	0.5722	1.0364
1000	0.8253	0.8009	0.7598	1.0653
2000	0.9560	0.9056	0.8813	1.0566
3000	1.0130	0.9890	0.9453	1.0551
4000	1.0282	1.0212	0.9890	1.0456
5000	1.0673	1.07506	1.002	1.0505

Tableau V.6 – temps de destockage des différentes méthodes par rapport au nombre de type produit pour un stockage avec heuristique et un lot de taille 5

V.4 Analyse et comparaison des résultats de simulation

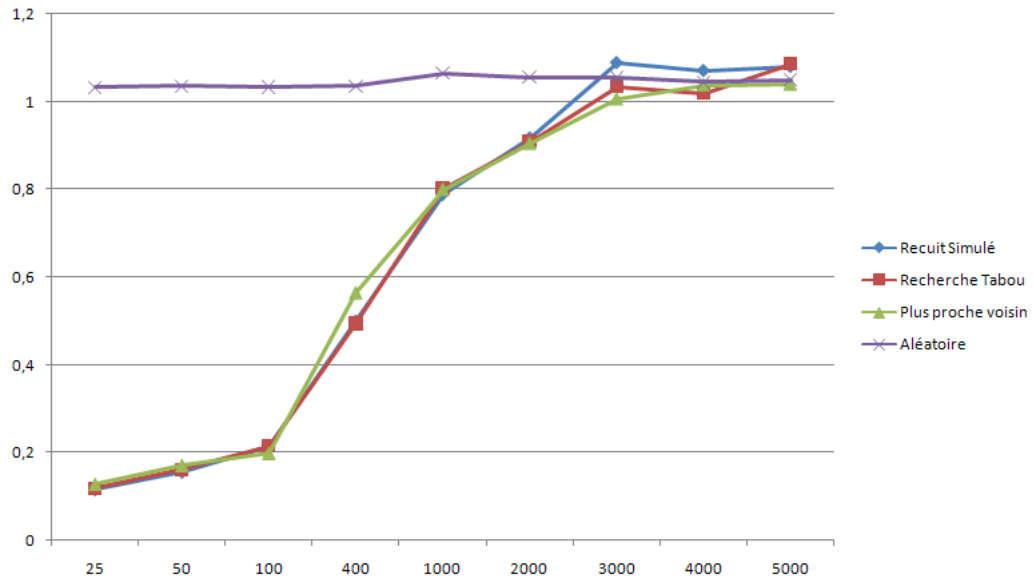


Figure V.2 – temps de destockage des différentes méthodes par rapport au nombre de type produit pour un stockage avec heuristique et un lot de taille 2

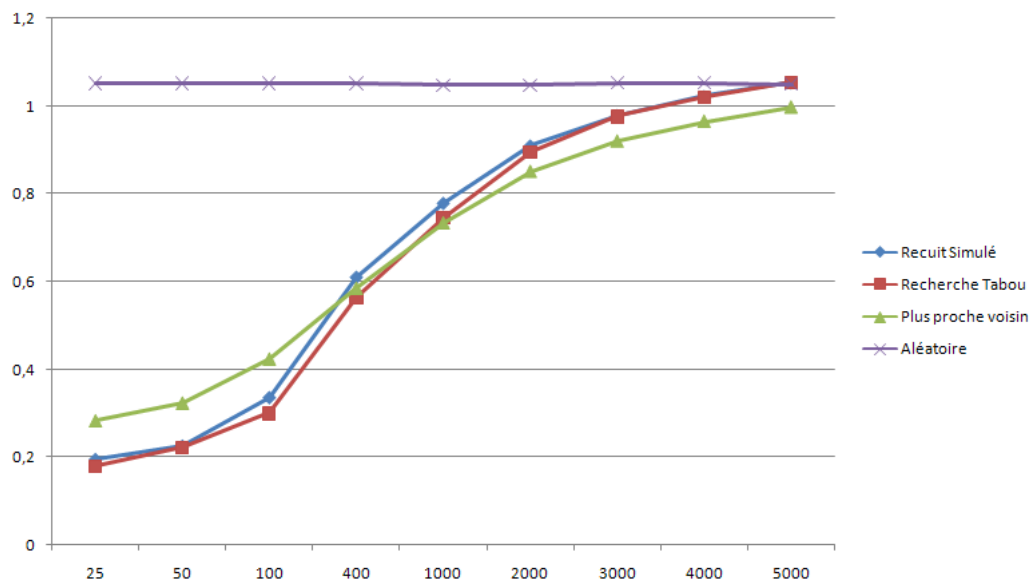


Figure V.3 – temps de destockage des différentes méthodes par rapport au nombre de type produit pour un stockage aléatoire et un lot de taille 5

V.4 Analyse et comparaison des résultats de simulation

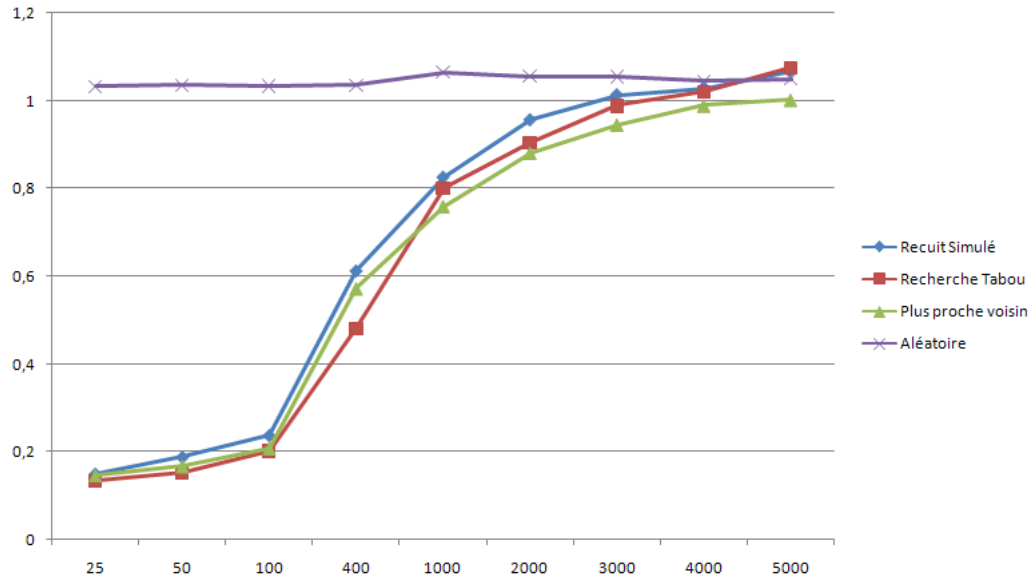


Figure V.4 – temps de destockage des différentes méthodes par rapport au nombre de type produit pour un stockage avec heuristique et un lot de taille 5

Taille du lot = 200			
N°type produits	Recuit Simulé	recherche tabou	plus proche voisin
25	0.2842	0.3934	0.1693
50	0.3488	0.4205	0.1834
100	0.4525	0.5133	0.2033
400	0.6807	0.6263	0.3216
1000	0.8324	0.7642	0.5209
2000	0.9254	0.8862	0.6887
3000	0.9644	0.9447	0.7711
4000	0.9875	0.9812	0.8213
5000	1.0048	1.0267	0.8546

Tableau V.7 – temps de destockage des différentes méthodes par rapport au nombre de type produit pour un stockage aléatoire et un lot de taille 200

V.4 Analyse et comparaison des résultats de simulation

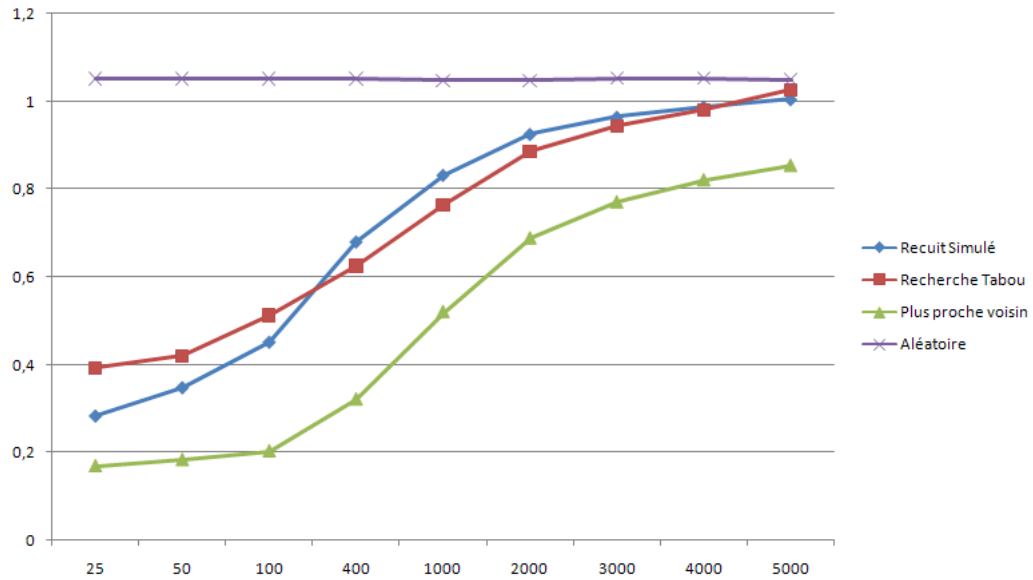


Figure V.5 – temps de destockage des différentes méthodes par rapport au nombre de type produit pour un stockage aléatoire et un lot de taille 200

Taille du lot = 200			
N°type produits	Recuit Simulé	recherche tabou	plus proche voisin
25	0.1537	0.1672	0.1266
50	0.1845	0.1951	0.1614
100	0.2295	0.2489	0.1998
400	0.5568	0.5607	0.3962
1000	0.8557	0.8099	0.6819
2000	0.9040	0.8748	0.8216
3000	0.9290	0.9423	0.8825
4000	0.9760	0.9826	0.9631
5000	0.9649	0.9755	0.9385

Tableau V.8 – temps de destockage des différentes méthodes par rapport au nombre de type produit pour un stockage avec heuristique et un lot de taille 200

V.4 Analyse et comparaison des résultats de simulation

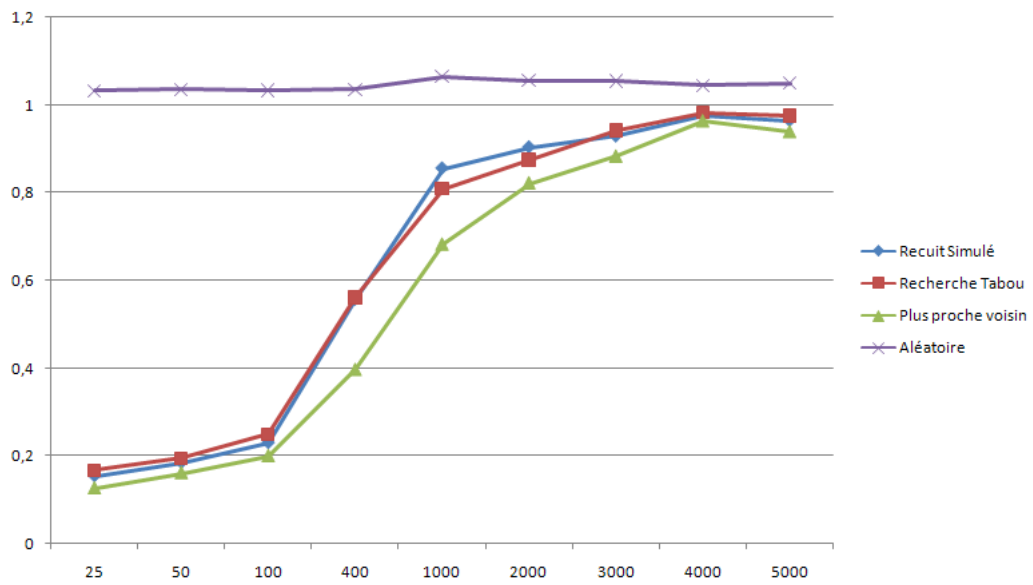


Figure V.6 – temps de destockage des différentes méthodes par rapport au nombre de type produit pour un stockage avec heuristique et un lot de taille 200

Conclusion

Dans ce chapitre, nous avons adapté les métaheuristiques recuit simulé et recherche tabou pour minimiser le temps de destockage dans le système AS/RS à convoyeur gravitationnel ; ensuite, nous avons présenté les différentes expérimentations réalisées pour vérifier l'efficacité de ces méthodes sur notre problème. Ceci nous a menés à conclure que les deux metaheuristiques donnent de meilleurs résultats que le plus proche voisin quand la taille du lot est moyenne et le nombre de produits différents est moyen, ou petit, lorsque le nombre des produits différents et la taille du lot augmente ; la méthode du plus proche voisin donne de meilleurs résultats

Conclusion générale et perspectives

Depuis les débuts des années 50, les systèmes automatisés de stockage/déstockage sont très utilisés dans beaucoup de systèmes de production et de distribution, ce qui a mené au développement d'une grande variété de type d'AS/RS pour étreindre tout le domaine de leur fonctionnement et pour être employés pour tout genre de produits.

L'AS/RS à convoyeur gravitationnel (Flow Rack Automated Storage/Retrieval System) est très peu étudié. Ce type d'AS/RS est utilisé dans l'industrie agro-alimentaire où un seul type de produits, ou très peu de produits différents, y sont stockés. Les mêmes produits sont stockés dans les casiers et le déstockage se fait suivant la règle FIFO. Cette façon de l'utiliser limite considérablement son champ d'application. C'est pourquoi nous nous sommes intéressés à étudier ce type de système et d'implémenter de nouvelles stratégies d'utilisation pour augmenter ses performance. L'un des inconvénients des AS/RS à convoyeur gravitationnel est le temps de cycle qui est plus long dans la plupart des cas que celui des AS/RS à charge unitaire classiques. Nous nous sommes proposés d'adapter deux métaheuristiques : recuit simulé et recherche tabou aux AS/RS à convoyeur gravitationnel afin de minimiser les temps de déstockage des produits.

Pour l'étude de ce système, nous avons développé un simulateur dédié au système AS/RS à convoyeur gravitationnel en langage JAVA sous l'environnement Eclipse, où nous avons adapté et implémenté les méthodes recuit simulé, recherche tabou, le plus proche voisin, le stockage aléatoire, le déstockage aléatoire et l'heuristique de stockage/déstockage développé par Sari 2003, que nous avons simulés pour analyser les performances du système pour chaque méthode utilisée.

L'analyse des résultats de la simulation nous a permis de déduire que :

- le système converge vers l'aléatoire, quelque soit la méthode utilisée si le nombre des

produits différents augmente.

- si la taille du lot est petite les trois méthodes recuit simulé, recherche tabou et plus proche voisin sont presque identiques
- si la taille du lot est moyenne et le nombre de produits différents petit, alors les métaheuristiques donnent de meilleurs résultats que le plus proche voisin ; si le nombre de produit augmente la méthode du plus proche voisin donne de meilleurs résultats.
- dans le cas ou la taille du lot est grande, la méthode du plus proche voisin donne les meilleurs résultats

Au cour de ce travail nous avons constaté que, dans certains cas, les métaheuristiques ne donnent pas des resultats satisfaisants par rapport à la méthode du plus proche voisin et, dans d'autres cas, la méthode du plus proche voisin ne donne pas de bons résultats par rapport aux métaheuristiques ; nous proposons alors comme perspective de combiner les méthodes pour tirer profit du meilleur de chacune de ces méthodes et d'améliorer le simulateur en implémentant d'autres métaheuristiques, et de développer et d'intégrer d'autres systèmes AS/RS dans le simulateur pour étudier leur comportement et efficacité.

Publication Internationale

Metaheuristic based control of a flow rack automated storage retrieval system

Hakim Nadir Bessenouci · Zaki Sari · Latéfa Ghomri

Received: 28 June 2010 / Accepted: 30 June 2010
© Springer Science+Business Media, LLC 2010

Abstract This paper presents two metaheuristic algorithms, namely taboo search and simulated annealing, applied to the control of a flow rack automated storage retrieval system (AS/RS). These metaheuristic algorithms are developed to control the retrieval machine of the AS/RS in order to minimize the retrieval cycle time. Results of these metaheuristic algorithms are compared to classical heuristics and analytical models found in literature (Gaouar et al. in MOSIM 2006, Rabat, Maroc, avril 2006; Sari et al. in Int J Adv Manuf Technol 25:979–987, 2005; Ghomri & Sari in Conception et Production Intégrées, CPI'2009, Fes, Maroc, 19–21 Octobre 2009). These heuristics were developed to control the storage and retrieval of items in the AS/RS. On the other hand, analytical models were conceived to bring a bottom line for comparison of different control techniques. To carry out this comparative study, simulations were performed on a wide range of system configurations. In order to validate metaheuristic results, a sensitive study on their parameters was achieved, and the best parameters were selected for comparative study.

Keywords Automated storage retrieval system · Flow rack AS/RS · Metaheuristics · Simulated annealing · Taboo search · AS/RS control

H. N. Bessenouci · Z. Sari (✉) · L. Ghomri
Automatic control laboratory, Aboubekr Belkaid University
of Tlemcen, Tlemcen, Algeria
e-mail: z_sari@mail.univ-tlemcen.dz

H. N. Bessenouci
e-mail: Hakim.Bessenouci@mail.univ-tlemcen.dz

L. Ghomri
e-mail: ghomri@mail.univ-tlemcen.dz

Introduction

Automated storage and retrieval systems (AS/RS) have been widely used not only as alternatives to traditional warehouses, but also as a part of advanced manufacturing systems (Lee 1997). Improved inventory management and control, increased storage capacity to meet long-range plans, quick response to locate/store/retrieve items, and reduced labour cost due to automation are among the major advantages provided by AS/RS (MHI 1977; Allen 1992).

The main components of an AS/RS are racks, storage/retrieval (S/R) machines, aisles, pickup/drop-off (P/D) stations, and pick positions. Racks are typically metal structures with locations that can accommodate loads (e.g., pallets) that need to be stored. Automated storage and retrieval machines can autonomously move, pick up and drop off loads. Aisles are formed by the empty spaces between the racks, where the storage/retrieval (S/R) machines can move. A pickup/drop-off (P/D) station is a location where retrieved loads are dropped off, and where incoming loads are picked up for storage. Pick positions (if any) are places where people are working to remove individual items from a retrieved load before the load is sent back into the system (Roodbergen and Vis 2009). There exist several types of AS/RS that can handle items of different size and weight. These several types include unit-load, mini-load, man-on-board, deep-lane, automated item-retrieval system, and flow-rack systems.

A flow rack AS/RS consists of a deep rack composed of a matrix of horizontal and vertical sloping bins, where each bin consists of several segments and each segment can store a single item. Each sloping bin is equipped with a gravitational conveyor consisting of free rolling wheels or cylinders at its base. A storage machine and a retrieval machine are placed at the rear and the front of the rack, respectively.

A restoring gravitational conveyor, which is placed on one side of the rack, is used to link the retrieval machine to the storage machine. The products are stored in sloping bins from the rear of the rack and they slide toward the front of the rack on the gravitational conveyor inside the bin until they reach the end of the queue of previously stored items. When an item is to be retrieved, if it is stored in the first segment of the bin (i.e., the nearest segment to the retrieval machine). The retrieval operation simply consists of picking the item from the bin and depositing it in the delivery station. If the desired item is not stored in the first segment, then the retrieval operation consists of two phases: First, all items stored before the desired item are retrieved and delivered to the restoring conveyor. These items are transported on the restoring conveyor until they reach the storage face of the rack where they are picked up by the storage machine and stored back in the rack. Second, when the desired item reaches the first storage segment, it is then picked up by the retrieval machine and delivered to the deposit station.

Various simulation-based studies that analyze throughput performance of AS/RS exist in the literature (Sand 1976; Barrett 1977; Schwarz et al. 1978; Koenig 1980). These studies compare different operating policies for a given system configuration. Various analytical approaches to develop cost models for AS/RS have been proposed. Bozer and White (1980) present a design package that uses Zollinger's cost model (Zollinger 1975). Karaswa et al. (1980) propose a cost model for single command cycles.

There is extensive research in the area of dwell point of S/R machine. Bozer and White (1984) suggest static dwell point rules, although they provide no quantitative comparison of their performance. Egbelu (1991) presents a model for dynamic positioning of S/R machines with the objective of minimizing the expected travel-time. In their study, Hwang and Lim (1993) show that the formulation, proposed by Egbelu (1991) could also be applied to facility location problems. In another study, Egbelu and Wu (1993) compare the performance of several dwell point rules, adopted from Bozer and White (1984) and Egbelu (1991), using simulation. Peters et al. (1996) develop a closed form solution for dwell point location under a variety of AS/RS configurations. Chang and Egbelu present formulations for pre-positioning of S/R machines in order to minimize the maximum system response time (Chang and Wen 1997) and minimize the expected system response time (Chang and Egbelu 1997b) for multi-aisles AS/RS. Park (2001) developed an optimal dwell point policy for automated storage/retrieval systems with uniformly distributed racks. He proposed, for non-square-in-time racks, a closed form solution for the optimal dwell point in terms of the probability of the next transaction demand type: storage or retrieval.

Development of expected travel-time (i.e. average travel-time) models for a S/R machine is another research area. A

comparative study based on the expected travel-time of a S/R machine for randomized and dedicated storage policies has been presented by Hausman et al. (1976). The rack configuration has been assumed square in time (i.e. horizontal maximum travel-time is equal to vertical maximum travel time) with single and dual command cycles. An extension on the model of Hausman et al. (1976) has been proposed by Graves et al. (1977). They present analytical and empirical results for various combinations of alternative storage assignment rules and scheduling policies. Each alternative is compared on the basis of the expected travel-time of the S/R machine. Based on a continuous rack approximation approach, Bozer and White (1984) present expressions for expected cycle times of an AS/RS performing single and dual command cycles. Hwang and Lee (1990) present travel-time models, which include constant acceleration and deceleration rates with a maximum-velocity restriction. Chang et al. (1995) propose travel-time models that consider various travel speeds with known acceleration and deceleration rates. Chang and Wen (1997) extend the work presented by Chang et al. (1995) by investigating the rack configuration problem. Sarker and Babu (1995) presented a brief critical review and comparative study of some design aspects of AS/RS systems with a special emphasis on travel time models. Kouvelis and Papanicolaou (1995), presented explicit formulae for the expected single command cycle time for an optimally designed rack, for a two-class-based automated storage/retrieval system. Lee (1997) presented a stochastic analysis using a single-server queuing model to predict the cycle time (a mix of single and dual command) of a unit load AS/RS. Mansuri (1997) presented a computerised algorithm for cycle time computation and storage allocation for AS/RS under dedicated storage. Kulturel and Ozdemirel (1999) used the average travel time of S/R machines as the main performance measure, to compare two storage policies for AS/RS using simulation. Dallari et al. (2000) investigated the performance evaluation of a man-on-board AS/RS under different storage policies. The S/R machine travel time is derived for each storage policy as a function of the shape of the storage area, the number of picking points and the sequencing algorithm used. van Den Berg and Gademann (2000) presented a simulation study of an automated storage/retrieval system and examined a wide variety of control policies. For the class-based storage policy, they applied an algorithm which enables evaluation of the trade-off between storage space requirements and travel times. Ashayeri et al. (2002) presented an exact geometry-based analytical model which can be used to compute the expected cycle time for a storage/retrieval (S/R) machine, executing single commands, dual commands, or both, in a rack structure which has been laid out in pre-specified storage zones for classes of goods. Ghomri and Sari (2009) presented new models for single and dual cycle time of multi-aisle AS/RS. Their study was based on a continuous rack face and

aisle approximation. The closed form models they developed were compared to more complicated models for validation. Kouloughli et al. (2008) and Kouloughli et al. (2010) determined optimal dimensions of multi aisle AS/RS that minimize single and dual cycle time. Park (2006) and Park et al. (2006) studied performances of different AS/RS with class based storage policy. de Koster et al. (2006) and Yu and de Koster (2009) studied the design of an optimal rack of 3D compact storage under different storage policies. For a more detailed literature review, one can refer to Roodbergen and Vis (2009) who gave a comprehensive explanation of the current state of the art in AS/RS design, travel time estimation, storage assignment, dwell-point location, and request sequencing.

In industry, flow rack AS/RS is typically used for one or very few types of items, where each bin is dedicated to a particular item and the system operates based on the first-in-first-out rule. Sari et al. (2005) and Sari (2009) investigated the use of the system as an alternative of unit load AS/RS. They came with the conclusion that a flow rack AS/RS, when compared to an equivalent unit load system (i.e., same storage capacity, same number of storage and retrieval machines, and same height), requires less space and has a lower expected storage time. However the expected retrieval time of the flow rack is larger.

The objective of this work is to reduce the expected retrieval time in order that the flow rack AS/RS can be a good alternative to the unit load AS/RS. To achieve this objective, we developed two metaheuristic algorithms, namely taboo search and simulated annealing applied to the control of a flow rack automated storage retrieval system (AS/RS). These metaheuristic algorithms are developed to control the retrieval machine of the AS/RS in order to minimize the retrieval cycle time.

Description of flow rack AS/RS

A gravity flow-rack AS/RS is composed of only one rack, which consists of sloping bins. Products are loaded by a storage machine on the store face; they travel along sloping wheels or rollers to the pick face, to be retrieved by a retrieval machine. When an item is removed from a bin, the next item automatically rolls to the front of the rack. Both the retrieval and storage machines can travel on a x-y plane to reach any bin on the rack. A drop-off station and a pickup station are located at the store face and the pick face of the rack, respectively. A restoring conveyor is used to link the retrieval machine to the storage machine. As shown in Fig. 1, the rack, which consists of bins, has a length L , a height H and a depth D . The rack has N_l bins in each row and N_h bins in each column. Each bin has M storage segments, numbered from 0 to $M-1$. Each segment has a storage

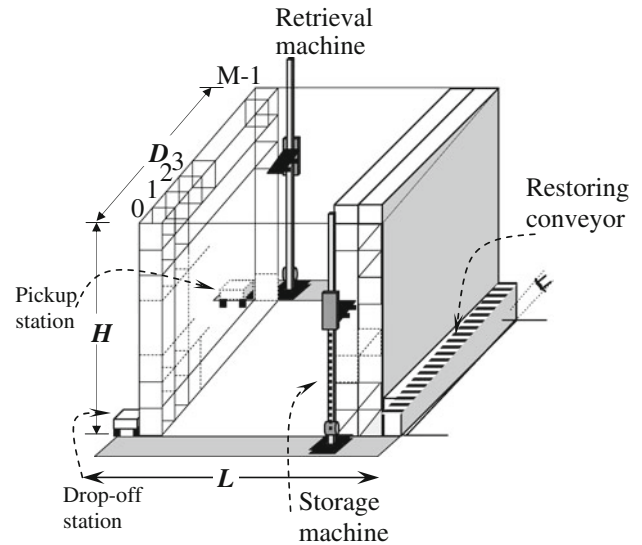


Fig. 1 Configuration and components of a flow rack AS/RS

capacity of one item. The length, height and depth of each segment are $l = L/N_l$, $h = H/N_h$ and $d = D/M$, respectively. Segments with the same rank form a layer. The storage and retrieval machines have the same horizontal speed, V_h , and the same vertical speed, V_v . Therefore, the travel times, th , tv , $t'h$ and $t'v$, can be calculated as follows:

$$\begin{aligned}
 th &= L/V_h \\
 tv &= H/V_v \\
 t'h &= l/V_h = th/N_l \\
 t'v &= h/V_v = tv/N_h
 \end{aligned}$$

When retrieving an item, the retrieval machine starts by taking out of rack, one by one, all the items which precede it in the bin. It transports them to the restoring conveyer, and finally takes out of stock the requested item. The restoring conveyer transports the items from the store face to the pick face, so that the storage machine replaces them in the system. The storage and retrieval machines move simultaneously in the horizontal and vertical directions. This displacement, known as Tchebychev displacement, allows a faster service of the machines.

Average storage and retrieval time of a flow rack AS/RS

The storage time model that we present in this section was developed by Sari et al. (2005). It is similar to the single travel time model developed by Bozer and White (1984) for unit load AS/RS. The storage and retrieval machines move with same speeds on two parallel rectangular surfaces of the same dimensions.

For a storage operation, a storage machine in a flow rack AS/RS operates exactly the same way as a S/R machine in a unit load AS/RS. Therefore, results of Bozer and White

(1984), which developed the expected cycle time expressions for a unit load AS/RS, may be used to represent the single cycle expected travel time as:

$$E(SC) = T \left(\frac{b^2}{3} + 1 \right), \quad \text{where } T = \max(t_v, t_h)$$

$$b = \min\left(\frac{t_v}{T}, \frac{t_h}{T}\right) \quad \text{and} \quad t_v = N_h t'_v t_h = N_l t'_h \quad (1)$$

The expected retrieval time for flow-rack AS/RS as derived by Sari et al. (2005) can be written as:

$$E(RC) = T \left(\frac{b^2}{3} + 1 \right) \left(\frac{1}{4} + \frac{1}{2} \rho \cdot M \right) \quad \text{for } \frac{1}{M} \leq \rho \leq 1 \quad (2)$$

$$E(RC) = \frac{3}{4} T \left(\frac{b^2}{3} + 1 \right) \quad \text{for } \rho < \frac{1}{M} \quad (2.1)$$

Storage and retrieval heuristic

This heuristic has been developed for the flow rack AS/RS by Sari (2003), in order to reduce the retrieval time. The storage procedure has three hierarchy priorities as shown below, while the retrieval procedure is based on the nearest neighbor heuristic.

- Storage procedure

Priority 1: store behind an identical product,

If a single bin, **then** store

Else If several bins **then**

Store in the bin where there is only one type of product.

If a single bin, **then** store

Else if several bins **then**

Store in bin nearest to the delivery station

If a single **then** store

Else if several bins **then**

Store in bin nearest to the dwell

point of the retrieval machine

Else

Store in bin where there is the largest number of products of this type.

If a single bin, **then** store

Else if several bins **then**.

Store in bin where there is the least different types of products

If a single bin, **then** store

Else if several bins **then**

Store in bin nearest to the delivery station

tion

If a single **then** store

Else if several bins **then**

Store in bin nearest to the

dwell point of the retrieval machine

Priority 2: store in an empty bin,

If a single bin, **then** store

Else if several bins **then**

Store in bin nearest to the delivery station

If a single bin **then** store

Else if several bins **then**

Store in bin nearest to the dwell point of the

retrieval machine

Priority 3: store in bin where there is the least number of products,

If a single bin, **then** store

Else if several bins **then**

Store in bin nearest to the delivery station

If a single bin **then** store

Else if several bins **then**

Store in bin nearest to the dwell point of the

retrieval machine

- Retrieval procedure

Retrieve product from layer nearest to the

retrieval machine

If a single bin, **then** retrieve

Else if several bins **then**

Retrieve from bin nearest to the delivery station

If a single bin **then** retrieve

Else if several bins **then**

Retrieve from bin nearest to the dwell point of

the retrieval machine

Metaheuristics

A metaheuristic is formally defined as an iterative generation process which guides a subordinate heuristic by combining intelligently different concepts for exploring and exploiting the search space. Learning strategies are used to structure information in order to find efficiently near-optimal solutions (Osman and Laporte 1996).

According to Blum and Roli (2003), the fundamental properties which characterize metaheuristics are:

- Metaheuristics are strategies that “guide” the search process.
- The goal is to efficiently explore the search space in order to find (near-) optimal solutions.
- Techniques which constitute metaheuristic algorithms range from simple local search procedures to complex learning processes.
- The basic concepts of metaheuristics permit an abstract level description.
- Metaheuristics are not problem-specific.
- Metaheuristics may make use of domain-specific knowledge in the form of heuristics that are controlled by the upper level strategy.

- Today, more advanced metaheuristics use search experience (embodied in some form of memory) to guide the search.

In this work we are intending to adapt two metaheuristic algorithms, namely: simulated annealing and Tabu search, for the control of a flow rack AS/RS in order to reduce the expected retrieval time

Simulated annealing

The Simulated Annealing algorithm has been introduced in the area of combinatorial optimization by Kirkpatrick et al. (1983). It relies on a model developed by Metropolis et al. (1953) for simulating the physical annealing process, where particles of a solid arrange themselves into a thermal equilibrium.

In pseudocode, the Simulated Annealing algorithm can be represented as follows (Bianchi et al. 2006):

Algorithm 1 Simulated Annealing

- 1 Initialize state x , temperature parameter T_1 and sample size N_1 ;
- 2 Set $x^* = x$ [x^* is the current approximation of the optimal solution];
- 3 for iteration $k = 1, 2, \dots$ do
- 4 select y randomly from $S(x)$;
- 5 compute sample average estimates $g_k(x)$ and $g_k(y)$ for the costs in x and y , respectively;
- 6 if $g_k(y) \leq g_k(x)$ then
- 7 set $x = y$;
- 8 else if $e^{\frac{g_k(x) - g_k(y)}{T_k}} \leq \text{uniform}[0,1]$ then
- 9 set $x = y$;
- 10 end if
- 11 compute a new current approximation x^* of the optimal solution;
- 12 update T_k to T_{k+1} ;
- 13 update N_k to N_{k+1} ;
- 14 end for

Therein,

- S is a finite set of feasible solutions (all not empty bins of AS/RS)
- x and y are feasible solutions from S ;
- T_1, T_2, \dots is a (usually decreasing) sequence of values for the temperature parameter; the update of the values T_k is $T_{k+1} = \alpha \cdot T_k$;
- the sets $S(x)$ form the pre-defined neighborhood structure: to each feasible solution
- $x \in S$, a set
- $S(x) \subseteq S \setminus \{x\}$ of {neighbor solutions} is assigned;

- uniform $[0,1]$ is a procedure selecting a uniformly distributed (pseudo-random) number from the interval $[0,1]$.

Tabu search

The method of search with tabu, or simply *tabu search* or *tabu method*, was formalized in 1986 by Glover (1986). Its principal characteristic is based on the use of mechanisms inspired by the human *memory* and employs a different approach to doing exploration: it keeps around a history of recently considered candidate solutions (known as the *tabu list*) and refuses to return to those candidate solutions until they are sufficiently far in the past. Thus if we wander up a hill, we have no choice but to wander back down the other side because we are not permitted to stay at or return to the top of the hill.

The simplest approach to Tabu Search is to maintain a **tabu list** L , of some maximum length l , of candidate solutions we have seen so far. Whenever we adopt a new candidate solution, it goes in the tabu list. If the tabu list is too large, we remove the oldest candidate solution and it is no longer taboo to reconsider. Tabu Search is usually implemented as a variation of Steepest Ascent with Replacement. In the version below, we generate n tweaked children, but only consider the ones which are not presently tabu (Luke 2010).

In pseudocode, the Tabu Search algorithm can be represented as follows (Bianchi et al. 2006):

Algorithm 2 Tabu Search

- 1 Generate a starting current solution x
- 2 Initialize the tabu lists
- 3 for iteration $k = 1, 2, \dots$ do
- 4 Set $A(x, k) = \{y \in S(x) | T(x, k)\}$
- 5 Set $x = \arg \min_{y \in A(x, k)} g(y)$
- 6 Update the tabu lists and the aspiration criteria
- 7 end for

Therein,

- $A(x, k)$ is the set of solutions among which the new current solution is chosen at iteration k
- $S(x)$ is the set of neighbors of x
- $T(x, k)$ is the set of tabu that moves at iteration k

Simulation study

To test the effectiveness of the proposed methodology, simulation experiments are carried out using a program developed in 'JAVATM'. The AS/RS considered for analysis has $N_l=20$ rows, $N_h = 20$ columns and $M = 10$ storage segments in each bin (or 10 layers). So a total number of storage spaces $N = 4000$. The centre distance between two adjacent bins in a row or column is 1 unit. The travel times $t'_h = t'_v = 0.01$.

Table 1 Sensitivity analysis results of simulated annealing

	t_i	t_f	α	S	$E(RC)$	$t_c(ms)$
	36	0.1	0.5	7	0.2969	0.3658
	44	0.1	0.5	7	0.2835	0.3874
	52	0.05	0.5	11	0.2788	0.4403
	44	0.1	0.6	7	0.2782	0.5513
	52	0.05	0.5	7	0.2780	0.3341
	36	0.1	0.5	11	0.2773	0.6086
	52	0.1	0.6	7	0.2774	0.3428
	44	0.05	0.5	10	0.2767	0.6041
	40	0.1	0.5	10	0.2758	0.4986
	52	0.1	0.5	10	0.2751	0.3822
	36	0.1	0.8	10	0.2217	1.0497
	28	0.001	0.6	10	0.2215	0.6049
	28	0.1	0.8	10	0.2214	1.2093
	44	0.001	0.6	10	0.2209	0.9765
	44	0.005	0.7	10	0.2207	1.0323
	44	0.05	0.7	11	0.2206	0.8971
	28	0.01	0.7	9	0.2205	1.0516
	40	0.01	0.7	10	0.2205	1.0860
	28	0.001	0.9	9	0.1932	2.8835
	40	0.001	0.9	7	0.1930	3.1258
	28	0.005	0.9	13	0.1917	5.1319
	28	0.005	0.9	11	0.1907	4.4331
	28	0.05	0.9	10	0.1906	2.7439
	28	0.01	0.9	9	0.1903	3.1773
	28	0.01	0.9	10	0.1902	3.5843
	44	0.005	0.9	11	0.1897	4.1399
	52	0.01	0.9	10	0.1895	2.6037
	52	0.001	0.9	11	0.1893	3.6943
	36	0.001	0.9	13	0.1864	4.5590
	28	0.001	0.9	10	0.1861	3.2445
	44	0.001	0.9	13	0.1857	5.1016
	40	0.005	0.9	11	0.1856	3.8967
	36	0.001	0.9	9	0.1855	3.6923
	28	0.001	0.9	13	0.1853	4.1399
	36	0.01	0.9	13	0.1850	3.2537
t_i Initial temperature	28	0.01	0.9	13	0.1849	4.6045
t_f Final temperature	40	0.001	0.9	13	0.1843	3.2662
α Cooling coefficient	52	0.001	0.9	13	0.1831	4.2756
S Thermodynamic balance	40	0.001	0.9	11	0.1812	3.2186
$E(RC)$ Mean retrieval time	52	0.001	0.9999	26	0.1810	6951.6
$T_c(ms)$ Computation time for one retrieval						

Sensitization study

In this section we will study the sensitivity of the results of the two metaheuristic (simulated annealing and tabu search) according to the setting parameters. The aim of this work is

the determination of the best parameters of the metaheuristics that grant the best results. To do this we conducted several simulations by varying the parameters, 625 case studies were simulated by simulated annealing and 125 by Tabu search. A summary of the results is given in Tables 1 and 2.

Table 2 Sensitivity analysis results of tabu search

	N_i	N_n	L	$E(RC)$	$t_c(ms)$
	1300	5	1	0.3137	8,9044
	1000	20	1	0.3037	10,298
	100	5	1	0.3025	0,8837
	100	20	5	0.2979	2,9518
	1000	1	1	0.2947	4,7566
	1300	15	1	0.2924	13,426
	1000	5	5	0.2367	19,271
	1300	1	5	0.2365	16,818
	1000	10	5	0.2361	40,267
	1300	5	5	0.2360	26,397
	100	20	10	0.2359	5,8049
	100	1	15	0.2182	3,7220
	100	20	15	0.2172	9,1860
	100	15	15	0.2165	11,204
	100	5	15	0.2141	9,0188
	100	10	15	0.2092	11,810
	100	5	20	0.2060	8,0674
	100	10	20	0.2018	12,497
	100	1	20	0.2016	8,6268
	100	20	20	0.1993	17,358
	100	15	20	0.1983	11,387
	1000	15	15	0.1900	125,91
	1000	10	15	0.1892	78,947
	1000	5	15	0.1881	70,850
	1000	20	15	0.1850	136,03
	1000	20	20	0.1848	204,45
	1000	15	20	0.1839	163,15
	1000	1	20	0.1839	70,445
	1300	5	15	0.1837	61,943
	1000	1	15	0.1836	57,085
	1300	5	20	0.1836	81,376
	1300	10	20	0.1825	134,41
	1300	20	20	0.1824	251,41
	1300	1	20	0.1818	93,927
N_i Number of iterations	1000	5	20	0.1816	109,31
N_n Number of neighborhood	1000	10	20	0.1815	118,21
L Tabu list size	1300	15	20	0.1800	214,17
$E(RC)$ Mean retrieval time	1300	15	100	0.1785	6545.4
$T_c(ms)$ Computation time for one retrieval					

For simulated annealing, the minimal retrieval time is given when the temperature cooling coefficient $\alpha = 0.9$ as shown in Table 1. The other parameters has very little or no influence on the retrieval time. In order to verify the influence of α , we carried out a last simulation by taking $\alpha = 0.9999$ the retrieval time did not change significantly but the computation time exploded.

For tabu search, the optimal retrieval time is obtained when the Tabu List size is the largest, as shown in Table 2. The other parameters have a little influence on the optimal retrieval time. To verify the impact of the tabu list, we carried out a simulation by taking the tabu list size=100. The retrieval time did not change significantly but the computation time exploded.

Table 3 Retrieval time considering various retrieval methods and numbers of different products, random method for storage

No different product	SA	TS	NN	R
25	0.1952	0.1798	0.2830	1.0516
50	0.2259	0.2218	0.3233	1.0515
100	0.3363	0.3001	0.4236	1.0519
400	0.6105	0.5637	0.5848	1.0528
1000	0.7789	0.7466	0.7341	1.0483
2000	0.9097	0.8952	0.8511	1.0482
3000	0.9788	0.9763	0.9204	1.0529
4000	1.0222	1.0199	0.9656	1.0529
5000	1.0535	1.0536	0.9977	1.0505

Table 4 Retrieval time considering various retrieval methods and numbers of different products, heuristic method for storage

No different product	SA	TS	NN	R
25	0.1487	0.1344	0.1462	1.0335
50	0.1871	0.1536	0.1666	1.0360
100	0.2365	0.2025	0.2072	1.0348
400	0.6116	0.4813	0.5722	1.0364
1000	0.8253	0.8009	0.7598	1.0653
2000	0.9560	0.9056	0.8813	1.0566
3000	1.0130	0.9890	0.9453	1.0551
4000	1.0282	1.0212	0.9890	1.0456
5000	1.0673	1.0750	1.0020	1.0427

Simulation analysis

The aim of this section is to evaluate and compare the expected retrieval times of the flow rack AS/RS, obtained by both retrieval methods: tabu search and simulated annealing. These results are compared to storage and retrieval heuristics such as random, nearest neighbor heuristic and analytical model of Sari et al. (2005). For this, we conducted several simulations by varying the number of different products in the system. Tables 3 and 4 summarize the simulation results. Column 1 of these tables gives the number of different types of items in the system, while the other columns present the expected retrieval time obtained by simulated annealing (SA), tabu search (TS), nearest neighbor heuristic (NN), random retrieval (R) and analytical model (AM). All these results of Table 3 have been obtained using random storage. Random storage has been chosen to verify clearly the effectiveness of each retrieval methods. Whereas, Table 4 shows results obtained using the storage heuristic of Sect. 4.

Results and discussion

The numerical examples (Tables 3, 4) provide four general remarks regarding the choice of resolution method:

1. tabu search and simulated annealing give the best results when the number of different products is small or medium
2. when the number of different products is large, the retrieval time of all methods is almost identical to the random retrieval one.
3. storage heuristic gives the best results when the number of different products is small or medium
4. When the number of the different products is large, the heuristic storage doesn't provide an improvement of retrieval time.

This can be explained by the fact that when the number of different items gets very large, all storage spaces of the rack will have almost equal probability to be used for retrieval, which means that we are in the case of random retrieval.

In this kind of systems, it is generally stated that the total number of different products is at most about 20% of the total number of storage spaces N . In our case $N = 4000$, so the total number of different types of products should not exceed 800. If we consider the results in Tables 3 and 4, keeping in mind the last remark, we can state that simulated annealing, tabu search and nearest neighbor heuristic have similar performances with a little advantage to tabu search.

Conclusions

Automated Storage and Retrieval Systems (AS/RSs) are warehousing systems that are used for the storage and retrieval of products in both distribution and production environments. It is an important materials handling facility that offers flexibility to keep pace with the rapidly changing demands of manufacturing. Considering the limitations of the existing methods, we developed two metaheuristic algorithms namely, tabu search and simulated annealing, to control the retrieval machine of the AS/RS in order to minimize the retrieval cycle time. Because these methods are a global, randomised and structured search technique, it is able to provide solutions for improved performance of AS/RS. When the number of different products is small or medium tabu search and simulated annealing as a retrieval policy performs better than other heuristics currently being used.

References

- Allen, S. L. (1992). A selection guide to AS/R systems. *Industrial Engineering*, 24, 28–31.
- Ashayeri, J., & Heuts, R. M. (2002). A geometrical approach to computing expected cycle times for zone-based storage layouts in AS/RS. *International Journal of Production Research*, 40, 4467–4483.
- Ashayeri, J., Heuts, R. M., Valkenburg, M. W. T., Veraart, M. R., & Wilhelm, H. C. (2002). A geometrical approach to computing expected cycle times for zonebased storage layouts in AS/RS. *International Journal of Production Research*, 40(17), 4467–4483.
- Barrett, B. G. (1977). *An empirical comparison of high-rise warehouse policies for operator-controlled stacker cranes*. Rochester, NY: Logistics Research and Analysis, Eastman Kodak Company.
- Bianchi L., Dorigo M., Gambardella L. M., & Gutjahr W. J. (2006) *Metaheuristics in Stochastic Combinatorial Optimization: A survey*. IDSIA—Dalle Molle Institute for Artificial Intelligence Via Cantonale, Galleria 2, 6928 Manno Switzerland.
- Blum, C., & Roli, A. (2003). Metaheuristics in combinatorial optimization: Overview and conceptual comparison. *ACM Computing Surveys*, 35(3), 268–308.
- Bozer Y. A., & White, J. A. (1980). Optimum designs of automated storage/retrieval systems. In *Proceedings of the TIMS/ORSA Joint National Meeting*. Washington, DC.
- Bozer, Y. A., & White, J. A. (1984). Travel-time models for automated storage/retrieval systems. *IIE Transactions*, 16, 329–338.
- Chang, S. H., & Egbelu, P. J. (1997). Relative pre-positioning of storage/retrieval machines in automated storage/retrieval systems to minimize maximum system response time. *IIE Transactions*, 29, 303–312.
- Chang, D. T., Wen, U. P., & Lin, J. T. (1995). The impact of acceleration/deceleration on travel-time models for automated storage/retrieval systems. *IIE Transactions*, 27, 108–111.
- Chang, D. T., & Wen, U. P. (1997). The impact of rack configuration on the speed profile of the storage and retrieval machine. *IIE Transactions*, 29, 525–531.
- Chang, S. H., & Egbelu, P. J. (1997). Relative pre-positioning of storage/retrieval machines in automated storage/retrieval systems to minimize expected system response time. *IIE Transactions*, 29, 313–322.
- Dallari, F., Marchet, G., & Ruggeri, R. (2000). Optimisation of man-on-board automated storage/retrieval systems. *Integrated Manufacturing Systems*, 11, 87–93.
- de Koster, M. B. M., Le-Anh, T., & Yu, Y. (2006). Optimal storage rack design for a 3-dimensional compact AS/RS. *International Journal of Production Research* 46(6), 1495–1514.
- Egbelu, P. J., & Wu, C. T. (1993). A comparison of dwell point rules in automated storage/retrieval systems. *International Journal of Production Research*, 31, 2515–2530.
- Egbelu, P. J. (1991). Framework for dynamic positioning of storage/retrieval machines in an automated storage/retrieval system. *International Journal of Production Research*, 29, 17–37.
- Gaouar, N., Sari, Z., & Ghouali, N. (2006) Modélisation et implémentation d'une heuristique de stockage/déstockage pour un AS/RS à convoyeur gravitationnel dans l'environnement ARENA. *MOSIM 2006*, Rabat, Maroc, avril
- Ghomri, L., & Sari, Z. (2009). Modèle mathématique du temps de déstockage pour les AS/RS à convoyeur gravitationnel. *Conception et Production Intégrées, CPI'2009*, Fes, Maroc, 19–21 Octobre.
- Glover, F. (1986). Future paths for integer programming and links to artificial intelligence. *Computers & Operations Research*, 13, 533–549.
- Graves, S. C., Hausman, W. H., & Schwarz, L. B. (1977). Storage retrieval interleaving in automatic warehousing systems. *Management Science*, 23, 935–945.
- Hausman, W. H., Schwarz, L. B., & Graves, S. C. (1976). Optimal storage assignment in automatic warehousing systems. *Management Science*, 22(6), 629–638.
- Hwang, H., & Lim, J. M. (1993). Deriving an optimal dwell point of the storage/retrieval machine in an automated storage/retrieval system. *International Journal of Production Research*, 31, 2591–2602.
- Hwang, H., & Lee, S. B. (1990). Travel-time models considering the operating characteristics of the storage and retrieval machine. *International Journal of Production Research*, 28(10), 1779–1789.
- Karaswa, Y., Nakayama, H., & Dohi, S. (1980). Trade-off analysis for optimal design automated warehouses. *International Journal of Systems Sciences*, 11, 567–576.
- Kirkpatrick, S., Gelatt, C. D., & Vecchi, M. P. (1983). Optimization by simulated annealing. *Science*, 220(4598), 671–680.
- Koenig, J. (1980). Design and model the total system. *Industrial Engineering*, 12, 22–27.
- Kouloughli, S., Sari, Z., & Sari, T. (2008). Optimisation des dimensions d'un AS/RS multi allées pour un temps de simple cycle minimale. In *Proceedings of MOSIM'08*, Paris, France. ISBN. 978-2—7430-1057-7.
- Kouloughli, S., Sari, Z., & Sari, T. (2010). Optimisation des dimensions d'un AS/RS multi-allée basée sur un modèle analytique du temps de simple cycle. *Journal Européen Des Systèmes Automatisés (JESA)*, 44(2), 135–159.
- Kouvelis, P., & Papanicolaou, V. (1995). Expected travel time and optimal boundary formulas for a two-class-based automated storage/retrieval system. *International Journal of Production Research*, 33, 2889–2905.
- Kulturel, S., & Ozdemirel, N. E. (1999). Experimental investigation of shared storage assignment policies in automated storage/retrieval systems. *IIE Transactions*, 31, 739–749.
- Lee, H. F. (1997). Performance analysis for automated storage and retrieval systems. *IIE Transactions*, 29(1), 15–28.
- Luke, S. (2010). *Essentials of Metaheuristics*. Zeroth Edition.
- Mansuri, M. (1997). Cycle-time computation, and dedicated storage assignment, for AS/R systems. *Computers & Industrial Engineering*, 33, 307–310.

- Metropolis, N., Rosenbluth, A. W., Rosenbluth, M. N., Teller, A. H. & Teller, E. (1953). Equations of state calculations by fast computing machines. *Journal of Chemical Physics*, 21, 1087–1091.
- MHI. (1977). Considerations for planning and installing an automated storage/retrieval systems, Material Handling Institute, Inc., AS/RS Document-100 7M.
- Osman, I. H., & Laporte, G. (1996). Metaheuristics: A bibliography. *Annals of Operations Research*, 63, 513–623.
- Park, B. C. (2001). An optimal dwell point for automated storage/retrieval systems with uniformly distributed, rectangular racks. *International Journal of Production Research*, 39, 1469–1480.
- Park, B. C. (2006). Performance of automated storage/retrieval systems with non-square-in-time racks and two-class storage. *International Journal of Production Research*, 44, 1107–1123.
- Park, B. C., Foley, R. D., & Frazelle, E. H. (2006). Performance of miniload systems with two-class storage. *European Journal of Operational Research*, 170, 144–155.
- Peters, B. A., Smith, J. S., & Hale, T. S. (1996). Closed form models for determining the optimal dwell point location in automated storage and retrieval systems. *International Journal of Production Research*, 34, 1757–1771.
- Roodbergen, K. J., & Vis Iris, F. A. (2009). A survey of literature on automated storage and retrieval systems. *European Journal of Operational Research*, 194, 343–362.
- Sand, G. M. (1976). *Stacker crane product handling systems*. : Eastman Kodak Company.
- Sari, Z. (2003). *Modélisation, Analyse et Evaluation des Performances d'un AS/RS à Convoyeur Gravitationnel*. PhD Thesis, Tlemcen University, Algeria.
- Sari, Z. (2009). Performance evaluation of flow-rack and unit load automated storage & retrieval systems. *Conception et Production Intégrées, CPI'2009*, Fes, Maroc, 19–21 Octobre.
- Sari, Z., Saygin, C., & Ghouali, N. (2005). Travel time models for flow-rack automated storage and retrieval systems. *International Journal of Advanced Manufacturing Technology*, 25(9–10), 979–987.
- Sarker, B. R., & Babu, P. S. (1995). Travel time models in automated storage/retrieval systems: A critical review. *International Journal of Production Economics*, 40, 173–184.
- Schwarz, L. B., Graves, S. C., & Hausman, W. H. (1978). Scheduling policies for automatic warehousing systems: Simulation results. *AIIE Transactions*, 10, 260–270.
- van Den Berg, J. P., & Gademann, A. J. R. M. (2000). Simulation study of an automated storage/retrieval system. *International Journal of Production Research*, 38, 1339–1356.
- Wen, U. P., Chang, D., & Chen, S. P. (2001). The impact of acceleration/deceleration on travel-time models in class-based automated S/R systems. *IIE Transactions*, 33, 599–608.
- Yu, Y., & de Koster, M. B. M. (2009). Designing an optimal turnover-based storage rack for a 3D compact automated storage and retrieval system. *International Journal of Production Research*, 47(6), 1551–1571.
- Zollinger, H. A. (1975). Planning, evaluating and estimating storage systems. In *Proceedings of the Advanced Material Handling Seminar*, Purdue University, IN.

Dans ce travail nous présentons deux métaheuristiques, à savoir la recherche tabou et le recuit simulé, appliquées au système de stockage déstockage à convoyeur gravitationnel. Ces métaheuristiques sont mis au point pour contrôler la machine de déstockage de l'AS / RS afin de minimiser le temps de cycle de déstockage. Les résultats obtenus de ces métaheuristiques sont comparés à des méthodes heuristiques classiques et des modèles analyse [NG06] [ZS05]; [LG09]. Les résultats obtenue de ces métaheuristiques sont comparés à ceux obtenus de la littérature en appliquant des heuristiques classiques et des modèles analytiques pour le même problème ces heuristique sont développées pour le pilotage du stockage et destockage des articles dans l'AS/RS, par ailleurs, les modèles analytiques sont conçu afin d'apporter l'essentiel de la comparaison de différentes techniques de controles d'un AS/RS à convoyeur gravitationnel. pour effectuer cette étude comparative, des simulations sont effectuées pour des systèmes de différentes configurations,une étude de sensibilité des différents paramètres de notre système est effectuée. Ainsi les meilleurs paramètres sont sélectionnés pour l'étude comparative.

Mots clés : Système automatisé de stockage déstockage, AS/RS à convoyeur gravitationnel, Métaheuristiques, Recuit simulé, recherche tabou, pilotage d'un AS/RS

This work presents two metaheuristic algorithms, namely taboo search and simulated annealing, applied to the control of a flow rack automated storage retrieval system (AS/RS). These metaheuristic algorithms are developed to control the retrieval machine of the AS/RS in order to minimize the retrieval cycle time. Results of these metaheuristic algorithms are compared to classical heuristics and analytical models found in literature [NG06] [ZS05]; [LG09]. These heuristics were developed to control the storage and retrieval of items in the AS/RS. On the other hand, analytical models were conceived to bring a bottom line for comparison of different control techniques. To carry out this comparative study, simulations were performed on a wide range of system configurations. In order to validate metaheuristics results, a sensitive study on their parameters was achieved, and the best parameters were selected for comparative study.

Keywords : Automated storage retrieval system. Flow rack AS/RS. Metaheuristics. Simulated annealing. Taboo search . AS/RS control

äPÍä Ýí äÐÇ ÇáÚää ÇËäíä ää ÇáäíËÇÄÑÓËíß : ËÇáËÇä PääÇ ÈËØËíPääÇ Úái äÙÇä ÍÝÙ ÇáäæÇÍ . áPÍ ÄäìÒË äËíä ÇáØÑíPËíä áÇËÍBä Ýí ÄáÉ ÓÍË ÇáäæÇÍ ääÙÇä ÍÝÙ ÇáäæÇÍ ÈØÑPË ÇáÍÍÑËË, áBí ÈPáá ää æPË ÍæNË ÇáÓÍË. ÇáäËÇÆÌ ÇáäËÍÓá ÚáíäÇ ÈæÇÓØË äËíä ÇáØÑíPËíä Èä äPÇÑäÈäÇ ÈÇáäËÇÆÌ ÇáäËÍÓá ÚáíäÇ ÈØÑP BáÇÓíBíË BÇä PÍ Èä ÇäìÇÒäÇ Ýí æPË äÒì æÇáäææÐÌ ÇáÑíÇÓí [ZS05], citeGaouar06, [LG09]; áPËä ÇäìÇÒ äËíä ÇáØÑíPËíä äËÍÓíä ÚááíË ÇáÓÍË Ýí äÙÇä ÇáËÍÒíä Úä ÇáÓÍË ÈØÑíPË ÇáÍÍÑËË ááPíÇä ÈäÈä ÇáäPÇNË Pääì ÈÇäìÇÒ ÈNäÇäì ÈÌÑíËí ÍÍË Pääì ÈÈNäÍË Bä ÇáØÑP ÇáäÐBæNË ÓÇÈPÇ æPÍ Èä ÇäìÇÒ ÚÍË ÈÌÇNË áÄäÙäË äÍËáÝË ÇáËØBíäÇË æPÍ Pääì ÈÌÑÇÓË ÈÍÓíÓíË áÇ ÍËíÑ ÁÍÓä ÇáÚæÇäá ÇáÍÇÐÖË ÈÇáØÑíPËíä áäÈä ÇáÍÑÇÓË ÇáäPÇNäË.

Keywords : ÇáÄäÙäË ÇáÇæËæäÇËíBíË áÚááíË ÇáËÍÒíä æÇáÓÍË; ÄäÙÇäË ÍÝÙ ÇáäæÇÍ ÈØÑPË ÇáÍÍÑË; äíËÄæÑíÓËË NæBæíÓäæáí; NæØÇÑÒ ØËæ; ÇáËÍBä Ýí ÇáäÙÇä ÇáÄæËæäÇËíBí áÍÝÙ ÇáäæÇÍ ÈØÑPË ÇáÍÍÑË

Références bibliographiques

- [ALL92] S.L. ALLEN. A selection guide to as/r systems. *Industrial Engineering*, 24 :28–31, 1992.
- [Ask93] STANDRIDGE C. R. Askin, R. G. *Modelling and Analysis of Manufacturing Systems*. PhD thesis, 1993. [17](#)
- [BAPH96] J. S. Smith B. A. Peters and T.S. Hale. Closed form models for determining the optimal dwell point location in automated storage and retrieval systems. *International Journal of Production Research*, 34 :1757–1771, 1996. [19](#), [25](#)
- [BIK03] R. J. Graves A. ST. Onge B. I. Kim, S. S. Heragu. Realization of short cycle time in warehouse replenishment and order picking. *International Journal of Production Research*, 41 :349–364, 2003. [20](#)
- [CE97a] S. H. Chang and P. J. Egbelu. Relative pre-positioning of storage/retrieval machines in automated/retrieval systems to minimize maximum system response time. *IIE Transactions*, 29 :303–312, 1997. [19](#)
- [CE97b] S. H. Chang and P. J. Egbelu. Relative pre-positioning of storage/retrieval machines in automated/retrieval systems to minimize maximum system response time. *IIE Transactions*, 29 :313–322, 1997. [19](#)
- [DBF98] G. Draghici, N. Brinzei, and I. Filipas. La modélisation et la simulation en vue de la conduite des systèmes de production. *Les cahiers des enseignements francophones en Roumanie*, pages 110–129, 1998. [4](#)
- [DKY08] T. Le-duc De Koster, R. and Y. Yugang. Optimal storage rack design for a 3-dimensional compact as/rs. *International Journal of Production Research*, 46 :1495–1514, 2008.
- [] Johann. Dréo. Adaptation de la méthode des colonies de fourmis pour l’optimisation en variables continues, application en génie biomédical. *Thèse de Doctorat Université de Paris 12*, 2004.
- [Egb91] P. J. Egbelu. Framework for dynamic positioning of storage retrieval machines in an automated storage retrieval system. *International Journal of Production Research*, 29 :17–37, 1991. [19](#)

- [EW93] P. J. Egbelu and C. T. Wu. A comparison of dwell point rules in storage/retrieval systems. *International Journal of Productions Research*, 31 :2515–2530, 1993. [19](#)
- [FA05] C. Carbonne F. Amato, F. Basile. An approach to control automated warehouse systems. *Control Engineering Practice*, 13 :1223–1241, 2005. [20](#)
- [FD00] Ruggeri R. F. Dallari, G. Marchet. Optimization of man-on-board automated storage/ retrieval systems. *Integrated Manufacturing Systems*, 11 :87–93, 2000.
- [FEM04a] R. J. Graves F. Eldemir and C. J. Malmberg. Amélioration des performances d'un as/rs à convoyeur gravitationnel. *de magister, Université de Tlemcen Algérie*, 2004. [7](#), [20](#), [21](#), [33](#)
- [FEM04b] R. J. Graves F. Eldemir and C. J. Malmberg. New cycle time and space estimation models for automated storage and retrieval system conceptualization. *International Journal of Production Research*, 42 :4767–4783, 2004.
- [Glo89] F. Glover. Tabu search -part i. *ORSA journal on computing Tome 1*, pages 190–206, 1989. [36](#), [40](#)
- [Glo90] F. Glover. Tabu search -part ii. *ORSA journal on computing Tome 2*, pages 4–32, 1990. [36](#)
- [Her05] A. Hertz. Les métaheuristiques : quelques conseils pour en faire bon usage, in gestion de production et ressources humaines : méthodes de planification dans les systèmes productifs. *Presses Internationales de Polytechnique*, pages 205–222, 2005. [iii](#), [36](#), [37](#)
- [HH93] J. M. Lim H. Hwang. Deriving an optimal dwell point of the storage/retrieval machine in an automated storage/retrieval system. *International Journal of Production Research*, 31 :2591–2602, 1993. [19](#)
- [HH00] CS. Ko H. Hwang. A study on multi aisle system served by a single storage/ retrieval machine. *Int J Prod Res*, 26 :1727–1737, 2000. [19](#)
- [Hol92] J. H. Holland. adaptation in natural and artificial systems. *MIT press*, 2ed edition, 1992. [36](#)
- [IO96] G. Laporte I.H. Osman. Metaheuristics a bibliography. *Annals of Operations Research*, 63 :513–623, 1996. [34](#)
- [JD03] Eric Taillard Patrick Siarry Johann Dréo, Alain Petrowski. Métaheuristiques pour l'optimisation difficile. *Eyrolles*, 2003. [iii](#), [1](#), [37](#), [38](#), [39](#), [40](#), [41](#), [42](#)
- [JN53] Edgar G. Johnson and Alfred O. Nier. Angular aberrations in sector shaped electromagnetic lenses for focusing beams of charged particles. *Physical Review*, 91(1), jul 1953.
- [JP04] J.J. Dumery C. Merlaud J.P. Trichard J. Perrin, F. Binet. Automatique et informatique industrielle : Bases théoriques, méthodologiques et techniques. *Nathan Technique*, 2004. [8](#)

- [JPVdb99] A.J.R.M Gademan J. P. Van den berg. Optimal routing in an automated storage retrieval system with dedicated storage. *IIE transactions*, 31 :407–415, 1999. 20
- [JT84] J.A. White J.A. TompKins. Facilities planing. *John Wiley and sons*, 1984. 12
- [JW82] H.D. Kinney J.A. White. Storage and warehousing in handbook of industrial engineering, g. salvendy. *John Wiley and sons*, 1982. 16
- [KH08] H. Alla K. Hachemi. Pilotage dynamique d’un système automatisé de stockage/déstockage à convoyeur gravitationnel. *Journal Européen des Systèmes Automatisés*, 42 :487–508, 2008. 8
- [KUL85] R. A. KULWIEC. Material handling handbook. *John Wiley and Sons, Second Edition*, 1985.
- [Law07] A. M. Law. Simulation modeling and analysis. *Mcgraw-Hill International*, 4, 2007. 46
- [Lee92] M. K. Lee. A storage assignment policy in man-on-board automated storage/retrieval system. *International Journal of Production Research*, pages 2281–2292, 1992. 20
- [Lee96] Schaefer S. K. Lee, H. F. Retrieval sequencing for unit-load automated storage and retrieval system with multiple openings. *International Journal of Production Research*, 34 :2943–2962, 1996. 20
- [Lee97] H. F. Lee. Performance analysis for automated storage and retrieval systems. *IIE Transactions*, 29 :15–28, 1997.
- [LG08] A. Guezzan T. Sari L. Ghomri, Z. Sari. Modèle analytique du temps de simple cycle d’un as/rs multi allées. *Actes des 7e conférence internationale de modélisation et simulationsMOSIM’08, Paris,, 2008. 19*
- [LG09] Z. Sari L. Ghomri. Modèle mathématique du temps de déstockage pour les as/rs à convoyeur gravitationnel. conception et production intégrées,. *CPI’2009*, 2009. 82, 83
- [MD05] M. P. Fanti M. Dotoli. A coloured petri net model for automated storage and retrieval systems serviced by rail-guided vehicles : a control perspective. *International Journal of Computer Manufacturing*, 18 :122–136, 2005. 20
- [MHH87] J. S. Shieh J. A. White M. H. Han, L. F. McGinnis. On sequencing retrievals in an automated storage/ retrieval system. *IIE transactions*, 19 :56–66, 1987. 19
- [NG06] N. Ghouali N. Gaouar, Z. Sari. Modélisation et implémentation d’une heuristique de stockage/déstockage pour un as/rs à convoyeur gravitationnel dans l’environnement arena. *MOSIM 2006*, 2006. 82
- [NM53] M. A. Teller E. Teller N. Metropolis, A. Rosenbluth. Equation of state calculations by fast computing machines. *J. chem Phys. tome 21*, pages 1087–1090, 1953. 36

- [NM97] P. Hansen N. Mladenovic. Variable neighborhood search. *Computers and Operations Research*, 024 :1097–1100, 1997. [36](#)
- [N.S96] N.Singh. systems approach to computer integrated design and manufacturing,. *John Wiley ans sons*, 1996. [10](#)
- [Pir02] M. Pirlot. Métaheuristiques pour l’optimisation combinatoire : un aperçu général. dans j. teghem et m. pirlot, éditeurs, optimisation approchée en recherche opérationnelle - recherches locales, réseaux neuronaux et satisfaction de contraintes. *Hermès Science Publications, Paris*, pages 25–55, 2002.
- [Rac96] Zoran Racic. *’Etude et essais du spectromètre à plasma DYMIO de la mission (MARS 96)*. PhD thesis, 1996.
- [RDM77] A. Mungwattana R. D. Meller. Considerations for planning and installing an automated storage/retrieval systems, material handling institute. *AS/RS Document-100 7M*, 1977. [9](#), [16](#), [21](#)
- [RDM05] A. Mungwattana R. D. Meller. As/rs dwell-point strategy selection at high system utilization : a simulation study to investigate the magnitude of the benefit. *International Journal of Production Research*, 43 :5217–5227, 2005. [19](#)
- [Ree95] Colin Reeves. Modern heuristic techniques for combinatorial problems. *McGraw-Hill London*, 1995.
- [Sar97] Z. Sari. Introduction aux systèmes flexibles de production,. *Rapport interne*, 1997. [17](#)
- [Sar03] Z. Sari. Modélisation, analyse et évaluation des performances d’un as/rs à convoyeur gravitationnel. *Thèse de doctorat d’état, Université de Tlemcen Algérie*, 2003.
- [SCG01] Z. Sari, H. Chetto, and N. Ghouali. Expected retrieval time models of flow rack as/rs. *CPI 2001, Fes, Morroco*, 034 :1–16, 2001. [6](#), [26](#)
- [SD00] M. Rachline S. Devaux. Introduction à l’automatisme : Schneider electric. *Encyclopédie économie 3000, série haute technologie*, 2000. [8](#)
- [SHA98] R. E. SHANNON. Introduction to the art and science of simulation. *Proceedings of the 1998 Winter Simulation Conference*, pages 07–14, 1998. [44](#)
- [Tai98] E. Taillard. La programmation à mémoire adaptative et les algorithmes pseudo gloutons : nouvelles perspectives pour les métaheuristiques. *Thèse d’habilitation, Université de Versailles Saint-Quentin-en-Yvelines, Versailles, France*, 1998.
- [Tai02] E. Taillard. Principes d’implémentation des métaheuristiques. dans teghem j. et pirlot m., éditeur, optimisation approchée en recherche opérationnelle - recherches locales, réseaux neuronaux et satisfaction de contraintes. *Hermès Science Publications, Paris*, 34 :57–79, 2002.

- [VRM08] A. Mungwattana V. R. Muppani. A branch and bound algorithm for class based storage location assignment. *European Journal of Operational Research*, 189 :492–507, 2008.
- [WHH00] S. C. Graves W. H. Hausman, L. B. Schwarz. Optimal storage assignment in automatic warehousing systems. *Management Science*, 22 :629–638, 2000. [19](#)
- [YB80] J.A. White Y.A. Bozer. Optimum design of automated storage/retrieval systems. *TIMS/ORSA Joint National Meeting, Washington, DC*, 1980. [16](#)
- [YB84] J.A. White Y.A. Bozer. Travel time for automated storage/retrieval systems. *IIE transactions*, 16,n4 :329–338, 1984. [19](#), [24](#), [25](#)
- [YB94] M. S. Srinivasan M. M. Y.A. Bozer, M.S. Cho. Expected waiting times in single-device trip-based material handling system. *European Journal of Operational Research*, 75 :200–216, 1994.
- [ZS03] N. Ghouali Z. Sari, SAYGIN C. Eimpact of pickup/dropoff stations and restoring conveyor locations on retrieval time models of flow-rack automated storage and retrieval systems. *IEEE Transactions on Robotics and Automation*, 2003. [26](#)
- [ZS05] N. Ghouali Z. Sari, C. Saygin. Travel time models for flow rack as/rs. *international journal of advanced manufacturing technology*, 25 :979–987, 2005. [6](#), [9](#), [10](#), [12](#), [17](#), [18](#), [19](#), [21](#), [24](#), [25](#), [26](#), [27](#), [28](#), [33](#), [54](#), [82](#), [83](#)